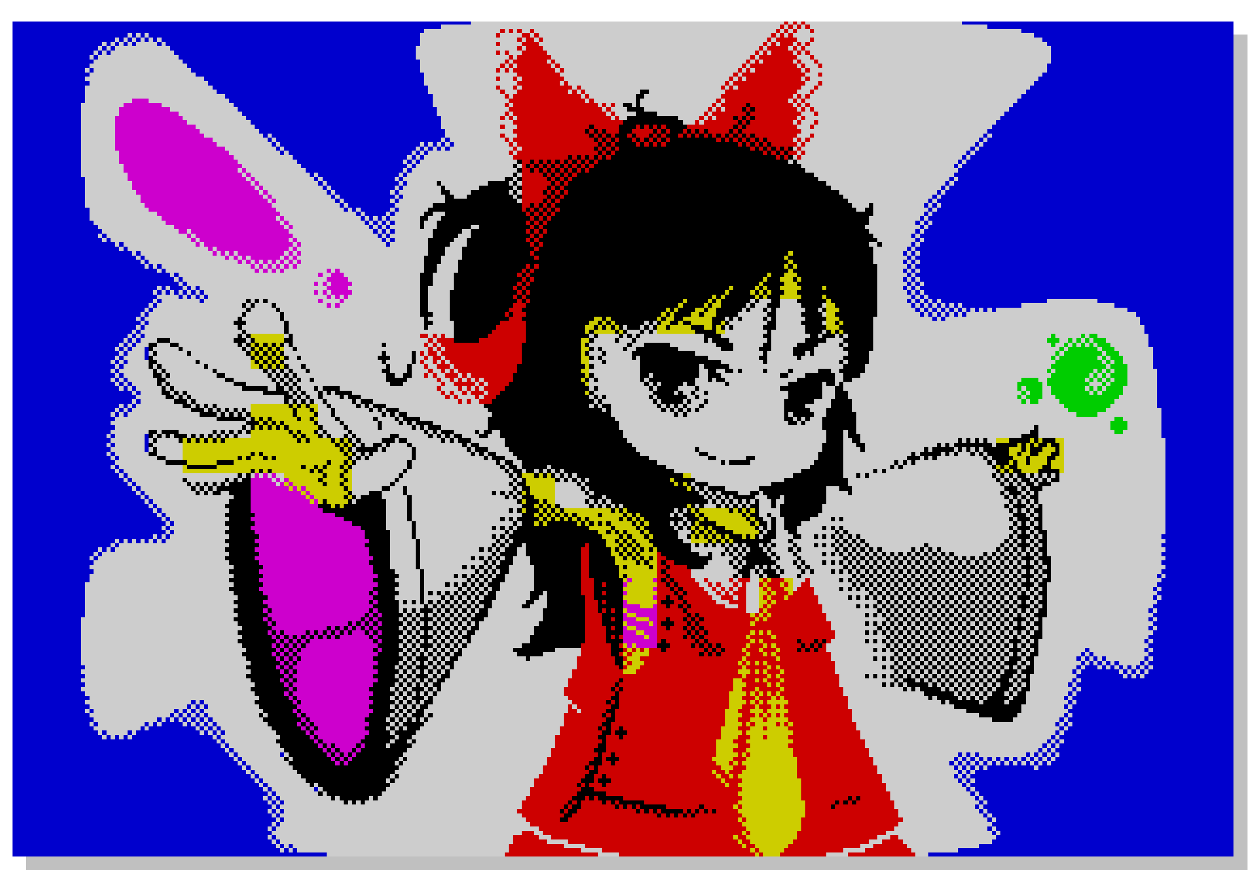




ZX Spectrum is Magic!

Herausgeber und für den Inhalt verantwortlicher:
Leszek Chmielewski, Prager Straße 92/11/12, 1210 Wien, Österreich
@Mail: retrozx@gmail.com
Klubkonto (Inhaber: Bernhard Lutz):IBAN: DE59 5486 2500 0000 5461 43
SWIFT-Code: GENODE6K, BIC-Code: GENODE61SUW
KTO.: 546143, BLZ: 54862500 (VR Bank Südpfalz, Sitz: Landau)

Ausgabe 228

1 Quartal 2012

Das Vorwort

http://www.womoteam.de/
http://spc.tlienhard.com/

Willkommen zu der Zeitschrift von Usern für User. Wir sind vor allem auf EURE Artikel angewiesen. Ich kann alleine keine (angepeilten) 24-32 Seiten füllen, so gerne ich es auch tun würde. Ehrenwort! Für eingeschickte Artikel gelten folgende Regeln:
Die Artikel müssen sich mit dem Spectrum, ZX81, SAM, Sprinter 2000 oder nahen Verwandten des Sinclair ZX Spectrum befassen, auch Artikel über passende Hardware und Software sind gerne gesehen.
MAC/PC Software: Nur wenn ausdrücklich direkt im Zusammenhang mit den eingangs erwähnten Computern. Sehr gerne: Crosscompiler, Emulatoren, Game Maker und dergleichen. Auf keinen Fall aber Remakes von Spielen alter Plattformen auf moderner Hardware.



Wieder mal hat die JH-Con stattgefunden, dieses mal war ich etwas früher als sonst angekommen und traf auf eine Gruppe von Leuten die an der Tür warteten. Da hat sich mal wieder mein Laserpointer bewährt mit dem ich den Organisatoren signalisierte, dass wir warteten. Einen hellen Lichtfleck des 200mW Lasers an der offenen Klassenzimmertür übersieht man nicht so leicht.
Es war jedenfalls wieder super und ich konnte dort „Quest for Witchcraft“ fertigstellen.

Eine Katastrophe folgt der nächsten: Zuerst ist meine Großmutter gestorben, dann am selben Tag ist mein PC kaputt geworden, und alle (!) E-Mails sind trotz Sicherung mit MozBackup verschwunden.
Gegen Problem 1 kann ich nichts machen. Problem 2 löste sich durch Neukauf des Mainboards, Prozessors und Speichers (wollte sowieso in dem Jahr aufrüsten) und Problem 3 ist wirklich schlimm, denn ich habe alle Daten verloren, obwohl ich am Tag zuvor ein Mail Backup gemacht habe. Leider sind dabei auch ältere Backups betroffen. Offenbar hat MozBackup die Mails meiner Konten nicht gesichert. Die Inbox Datei hatte immer eine Größe von 0 Bytes.
Leider waren damit auch alle eingeschickten Artikel weg.
Das neu-Aufsetzen des Rechners wurde leider notwendig. Mein alter AMD X2 4850e/4 GB RAM fror einfach von Zeit zu Zeit ein, so dass ich mehrmals Artikel verloren habe, die nicht abgespeichert wurden. Da das Einfrieren auch in BIOS passierte, lag es nicht am Windows. Laut Memtest86+ war der Speicher völlig okay, also blieben Prozessor und Mainboard übrig. Mir blieb kaum was anderes übrig als einen komplett neuen PC zu bauen. Der kann sich sehen lassen: AMD FX-6100 6-Kerner, 8 GB RAM, etc. Das schwerste war ein Mainboard mit Floppyanschluss zu

finden. Parallelport für GALEP IV habe ich mit einer PCI Karte realisiert, und erst später gemerkt, dass die Software unter 64 Bit nicht lauffähig ist (das gilt zum Glück aber nicht für den günstigeren und besseren (verarbeitungstechnisch wie auch was die Fähigkeiten betrifft) Wellon VP490). Später kam in den PC noch eine USB 2.0 Karte mit 2 internen USB Steckern (2x7-polige Pfosten) rein, weil zu wenige interne USB 2.0 Stecker vorhanden waren. So eine Karte habe ich lange gesucht, und bin froh sie gefunden zu haben.
Leider verzögerte das alles die Fertigstellung der aktuellen Ausgabe ganz beachtlich.

Ich habe aber die Gelegenheit genutzt um die Zeitschrift ein wenig zu verändern, aber keine Sorge, es sind nur dezente kosmetische Änderungen um an das Design von Wolfgang anzuschließen. Zusätzlich vollzog sich noch ein Umstieg auf Libre Office 3.5.0 RC3, wo weniger Fehler meine Arbeit geringfügig entlasten.

Ach ja, ich habe mich dazu entschlossen einige Spectrum & SAM Ersatzteile für Clubmitglieder günstig anzubieten. Fragt mal an wenn ihr etwas braucht, mit der benötigten Menge. Ich habe hier z.B. 20 Stück Z80 CPUs mit 6 MHz, geeignet für SAM Coupé. Für den SAM biete ich auch ein Ersatznetzteil an, welches kompakter ist und nicht nach ein paar Minuten zu fiepen beginnt (da die Meisten ohnehin Scart nutzen, hat das Netzteil keinen TV-Ausgang). Dafür akzeptiert es auch stromhungrigere Laufwerke. Zudem kann ich GALs für DivIDE, EPROMs beschreiben, etc. Natürlich habe ich nicht alle Ersatzteile aber wenn ich eine Quelle kenne, verweise ich gerne darauf. Nicht helfen kann ich derzeit bei ASIC/ULA.

Noch etwas: Offenbar versucht ein Abmahner von unserem Mitglied Norbert Opitz einen Betrag von mehreren tausend Euro heraus zu locken, mit dem falschen Vorwurf, dass er angeblich 2007 eine Karte auf die der Abmahner angeblich das Urheberrecht hat, als Wegweiser zum Spectrum-Treffen unerlaubt veröffentlicht hat. Dieser Vorwurf ist natürlich genauso lächerlich wie falsch. Aber auf jeden Fall Augen offen halten. Betrüger gibt es mehr als genug, und sie lauern wie die Haie auf unvorsichtige Nutzer, die sie als leichte Beute betrachten.

LCD-Leszek Chmielewski

Termine 2012

23.-25.03.2012

ZX Teamtreffen, ZX81, Termin bestätigt !

36145 Hofbieber (Mahlerts)

Tüftlertreffen der ZX81 Freunde. Hier wird noch richtig gebastelt!

Hallo Leute !

Der Termin für das Specci-Treffen 2012 in Wittenberg ist Sa./So. 25./26. August. Wir sind wieder im Kulturbund in der Lutherstraße 43a wie 2011. Jemand hatte den Vorschlag gemacht, die Tische ringsum an die Wand zu schieben, damit alles übersichtlicher und mehr Platz wird.
Bis zum nächsten Wiedersehen.

Tschüß

Norbert Opitz

Foreword-English

Again, the JH-Con has taken place, this time I arrived a little earlier than usual and met a group of people waiting at the door. I signaled the organizers that we were waiting. One bright spot of the 200mW LASER at the door of the classroom is not so easy to overlook.
Anyway, it was again great and I was able to finish my "Quest for Witchcraft"

A disaster is followed by the next: First, my grandmother died, then that day my PC got broken, and all e-mails have disappeared despite assurance with MozBackup (!).
I was forced to buy new mainboard, processor and memory (wanted to upgrade anyway in that year) but I lost all my Mails, although the day before I did a mail backup. Unfortunately, this also affects older backups. It appears that the MozBackup has not secured my stored mails, only the settings of accounts. The inbox file always had a size of 0 bytes.
So unfortunately,all sent and recieved mails were gone.
My old AMD X2 4850e / 4GB RAM simply froze from time to time, so I've lost several changes in the magazine that were not saved. Since the freeze happened in the BIOS too, it was not on Windows. According to the Memtest86+ memory was totally OK, so were left processor and motherboard. I had no choice but building a completely new PC and chosed the Bulldozer FX 6100 Hexacore with 8 GB RAM. The hardest part was finding a motherboard with floppy connector. Later came into the PC, a USB 2.0 card with 2 internal USB connectors (2x7-pin headers) purely because there are too few internal USB 2.0 plugs were. So a card I've been looking, and I'm glad I have found one.
Unfortunately, all this delayed the completion of the current issue is very substantial.
But I have taken the opportunity to change the magazine a little, but do not worry, there are connected only subtle cosmetic changes to meet the design of Wolfgang. Addition took place even switching to LibreOffice 3.5.0 RC3, where fewer errors slightly relieve my work.
Oh yes, I've decided to offer some Spectrum & SAM spare parts for club members. Ask if you need something, with the required amount. I have here for example 20 Z80 with a 6 MHz suitable for SAM Coupé. For the SAM I also offer a replacement power supply, which is compact and does not begin to whimpering after a few minutes (Since most people use anyway Scart, it has no TV output. It also accepts power-hungry drives. I also can write GALs for DivIDE, EPROMs, etc. Of course I do not have all the spare parts but if I know of a source, I refer to love. No help currently for ASIC / ULA.
One more thing: apparently a lawyer tried to sue one of our members Norbert Opitz to blackmail an amount of several thousand euros for the reason that he allegedly used 2007 a part of a map to describe the way to a Spectrum meeting. This accusation is, of course, as ridiculous as false. But in any case keep open eyes. Fraudsters, are more than enough around, and they lurk like sharks on unwary users to regard them as easy prey.

LCD Leszek Chmielewski

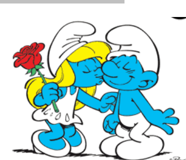
Neuigkeiten für unseren „Alten“

Jahreswechsel ist immer eine gute Zeit für News.

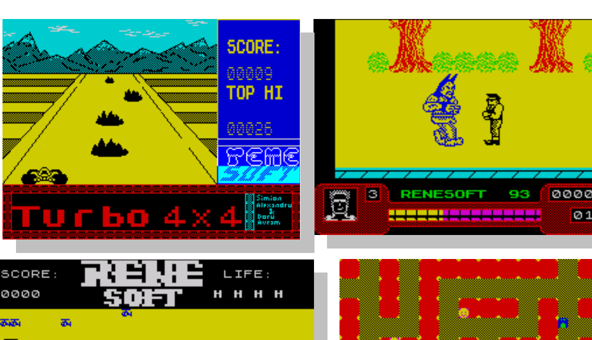
Der Weltenherr und die Schlümpfe

http://www.simion.co.uk/z80/





Alexandru Simion aus Rumänien hat seine alten ZX Spectrum Spiele online gestellt. Es sind Spiele, die vorher wohl unbekannt waren: Image, World Beyond, Master of World, Strumf (Smurf/Schlumpf), Arthur, JW Manision, Turbo 4x4, Dizzy – Magic Island, Aligator, Jungle Quest, Gun Combat, Runner Loader, Fly, Blooky,... Man kann zwar sicher nicht mit ruhigem Gewissen behaupten, dass es Meisterwerke sind, aber grottenschlecht sind sie auch nicht. Nur pixelpräzise Steuerung darf man bei den Sprites nicht erwarten.

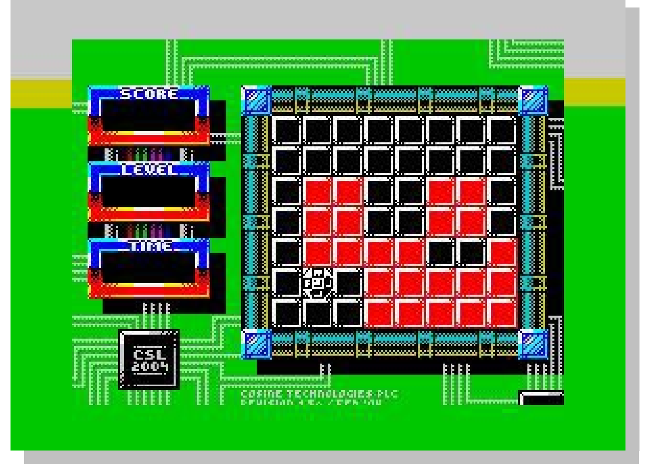






Reaxion

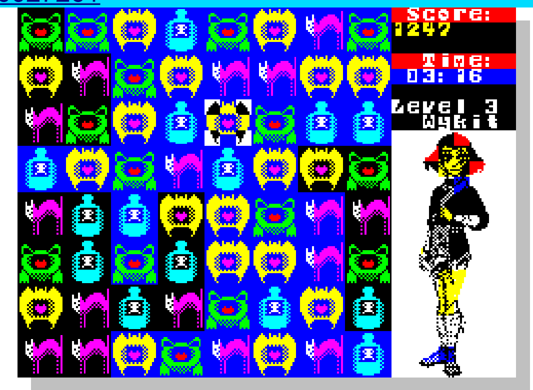
http://programbytes48k.wordpress.com/2011/12/20/preview-de-reaxion-para-zx-spectrum/



Reaxion galt schon fast als Vapourware, da es seit Jahren entwickelt wird, aber bislang nur Screenshots erschienen sind. Doch anscheinend geht die Arbeit an dieser Umsetzung vom C64 weiter. Auf der Replay 2011 wurde eine Vorschau vorgestellt. Und was soll ich sagen? Es schaut sehr gut aus.

Das ist Hexerei!

http://www.worldofspectrum.org/infoseekid.cgi?id=0027261



„Quest for Witchcraft“ ist ein Spiel welches ich exklusiv für Scene+ Diskettenmagazin geschrieben habe. Es wurde von „Jewel Quest“ inspiriert und wurde mit Boriels ZX BASIC Compiler geschrieben. Es bietet sehr schnelle Grafik und 7 Levels mit jeweils eigener Hexe die es zu besiegen gilt um in die Gilde der Witches 8 aufgenommen zu

werden. Vorsicht, das Spiel kann süchtig machen. Ein Demo ist auch da.

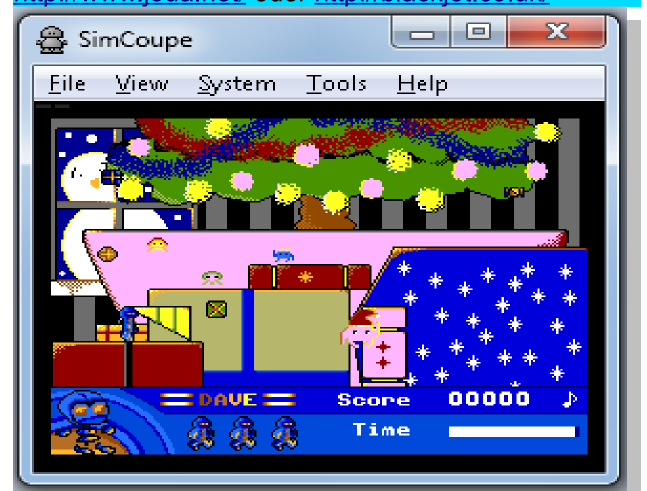
ScrPlus

http://sourceforge.net/projects/scrplus/

ScrPlus ist ein neuer Grafikkonverter mit benutzerfreundlicher GUI (behauptet jedenfalls der Programmierer), der 8x1 Attribute sowie ULA+ beherrscht. Es gibt Linux und Windows Versionen, und vielleicht mache ich was falsch, aber benutzerfreundlich ist das nicht, dass da mehrere Fenster aufgehen.

SAM Futter – Dave Invaders

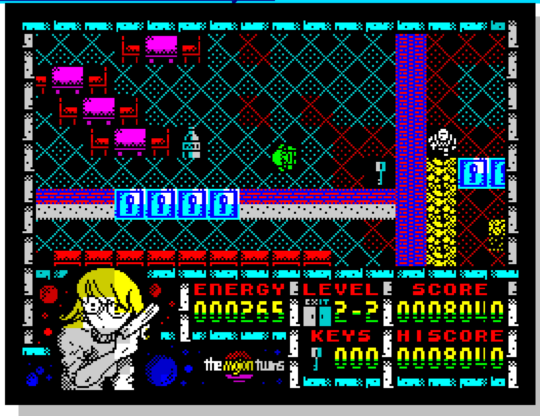
http://www.worldofspectrum.org/forums/showthread.php?p=598781&posted=1#post598781
http://www.joua.net/ oder http://blackjet.co.uk/



Es gibt wieder ein neues SAM Spiel, endlich! Es ist ein Plattformer nach bester Manic Miner Tradition, und benötigt einen SAM mit 512 Kb Speicher. Es wurde von Spud vollständig in Assembler geschrieben, und ist sein erstes Projekt für SAM. Nicht schlecht für ein Erstlingswerk, würde ich sagen.

Maritrini geht auf Monsterjagd

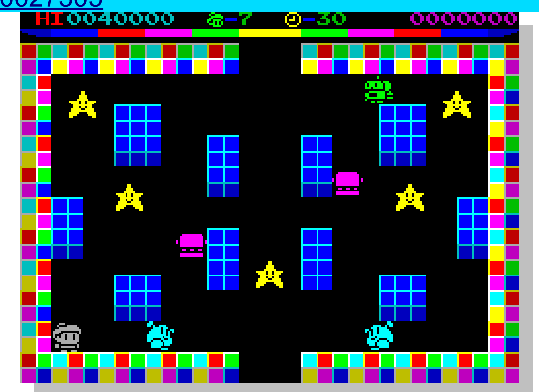
http://www.mojontwins.com/juegos_mojonos/maritrini-freelance-monster-slayer/



Und wieder machen Mojon Twins das, was sie am besten können: Den Spectrum mit neuem Stuff zu versorgen. Dieses mal haben sie einen scrollenden Gauntlet-Klon geliefert, der wie üblich optisch sehr ansprechend ist. Scrolling in einem Spiel das in C mit Z88dk geschrieben wurde, das sieht man nicht alle Tage.
Und wie bei Mojons Tradition ist, gibt es den C Sourcecode zum Download dazu.

Ist die Krone verloren?

http://www.worldofspectrum.org/infoseekid.cgi?id=0027305



Gloop Troops war schon ein Hit, und nun kommt der Level-Nachschub für Fans. Gloop Troops-The lost crown war zuerst nur in einer Spielesammlung für das iPhone erhältlich. Nun kann man das Spiel endlich als TAP/TZX herunterladen.

Gloop Troops die zweite

http://www.worldofspectrum.org/forums/showthread.php?t=37422



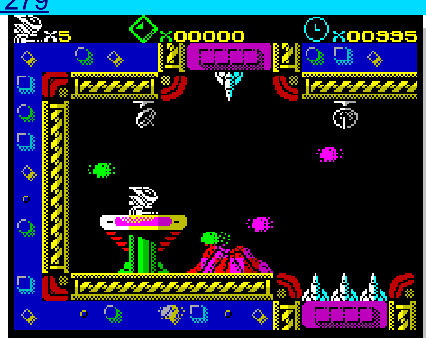
Wenn ich schon bei Gloop Troops bin. Gloop Troops II für Spectrum 128 nimmt Gestalt an, die Grafik und Conceptart sieht schon mal sehr vielversprechend aus.



Wir werden darüber auf jeden Fall berichten.

Sabernoid

http://www.worldofspectrum.org/infoseekid.cgi?id=0027279
http://www.worldofspectrum.org/infoseekid.cgi?id=0027279



Timmy hat seinen Cybernoid-Klon fertiggestellt. Das Spiel heißt „Future Looter“ und ist in C geschrieben (z88dk). Auch wenn es an das echte Cybernoid spielerisch nicht rankommt, grafisch ist es ihm auf jeden Fall überlegen.

Endloses Spielvergnügen

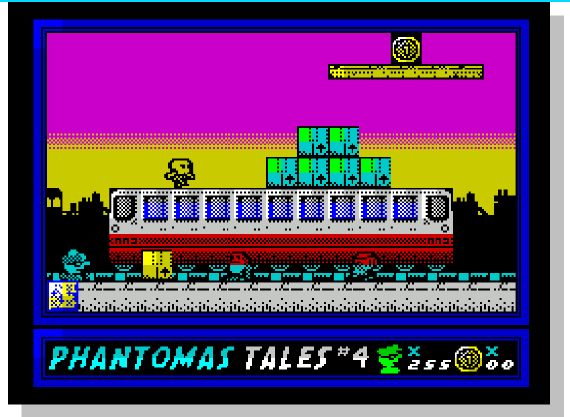
http://www.worldofspectrum.org/infoseekid.cgi?id=0027297
http://www.worldofspectrum.org/forums/showthread.php?t=37381



„Endless forms most beautiful“ ist das neueste Werk von R-Tape von Stonechat Productions. Es basiert auf einem 8x8 Spritegenerator mit Spiegelung, der scheinbar eine endlose Anzahl von unterschiedlichen Sprites zufallsbasierend erzeugen kann.

Alles gute zum 25-sten

http://www.mojontwins.com/juegos_mojonos/phantomas-tales-4-severin-sewers/



25 Jahre ist es her, dass das erste Phantomas-Spiel erschienen ist. Das haben Mojon Twins genutzt um zum Jubiläum ein neues Phantomas-Spiel herauszubringen. Phantomas Tales #4-Severin Sewers (Wo bleiben die Tales #2 und #3?). Das Spielprinzip bleibt

erhalten und Phantomas-typisch sind Sprünge je nachdem ob man nach unten oder oben drückt, kurz und hoch oder lang und niedrig. Ein besonderer Effekt ist die stimmungsvolle Beleuchtung in dunklen Screens.
Wieder mal ein spielenswerter Hit, den ich empfehlen kann. Mojon-untypisch ist keine nackte Haut zu sehen, wodurch das Spiel auch für konservative Gemüter geeignet ist.
Der C (z88dk) Sourcecode ist übrigens wie bei allen Mojon-Produktionen, ebenfalls herunterladbar.

Beatles vs. Willy

http://www.worldofspectrum.org/infoseekid.cgi?id=0027310



Willy meets the Beatles ist ein Spiel von Simon D. Lee für Spectrum 128, welches auf der beliebten Jet Set Willy Engine basiert.

I like Chopin

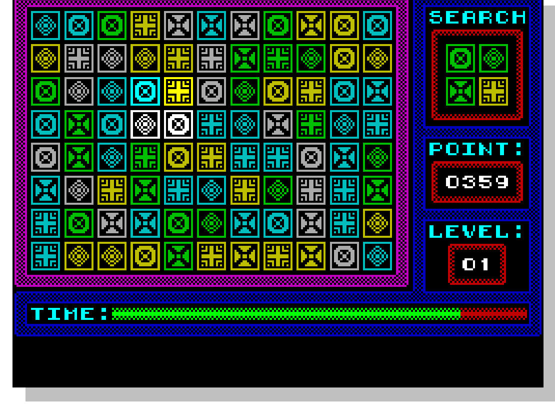
http://www.retroworks.es/php/game_en.php?id=10

Retroworks sind für seltenere aber qualitativ gute Releases bekannt. Die Jungs bleiben dem treu und haben nun nach längerer Releasepause Jinj 2 – Belmonte's Revenge veröffentlicht. Man läuft durch die Screens und sammelt Notenschlüssel auf. Technisch sehr gut umgesetzt. Wenn der blaue Hintergrund stört, kann man ihn im Menü abschalten, und das ist zu empfehlen, denn sonst wird man durch die Maske, die Attributclash vermeiden soll, ziemlich verwirrt.



Wo ist es denn, verd...?

http://www.worldofspectrum.org/infoseekid.cgi?id=0027299



Eher still und heimlich haben JeRrS aus Russland ihren neuesten Puzzler „4 from 88“ im TZX Format veröffentlicht. Man muss vorgegebene Muster von 4 zusammenhängenden Grafiken in einem Feld aus 88 Blöcken innerhalb eines vorgegebenen Zeitlimits finden. Es ist schwerer als man denkt, aber eine gute Spielidee.

Von MSX zu Spectrum

http://www.worldofspectrum.org/forums/showthread.php?t=36852

Glaubt man dem Thread auf WOS, und dem dort gezeigten Vorschauvideo, so setzt gerade die Gruppe Kralizec ihr MSX Spiel MajiKazo, welches sie 2006

geschrieben haben, für den Spectrum um. Der härteste Brocken ist wohl die Grafik und die Grafikroutinen (die noch IMHO etwas verbessert werden müssen), nachdem die Levels und alles Sonstige bereits vorhanden sein dürfte.

her, und nun haben sie das unfertige Spiel zum Download freigegeben. Fantastisch! Jetzt müsste man aber noch das Scrolling verbessern.



Wirklich bedauerlich, dass das Spiel nie fertiggestellt wurde.

Covertape für ???

http://www.worldofspectrum.org/infoseekid.cgi?id=0027307



Mojon Twins haben unfertige Demos, Prototypen, etc. zusammengestellt zu einem zweiseitigen Covertape #1, aber ich habe nicht die geringste Ahnung ob das für ein Magazin oder just for fun ist.

Mega Twins recovered

http://programbytes48k.wordpress.com/2011/12/25/mega-twins-demo-el-regalo-del-dia/

Mega Twins war wie z.B. auch Toki, ein Projekt das nie auf den Markt kam. Vor kurzem wurde auf eBay eine Auktion des Entwicklers abgewickelt, wo 3“ Disks mit dem unfertigen Spiel versteigert wurden. TZX Archive bekam den Zuschlag, stellte die Daten wieder

Flucht vom heiligen Turm

http://www.worldofspectrum.org/infoseekid.cgi?id=0027226
http://www.worldofspectrum.org/infoseekid.cgi?id=0027227



Dieses Puzzlespiel erschien zusammen mit einem Editor. An sich kein übles Spiel trotz primitiv-Grafik. Man muss Gegenstände aufsammeln und in richtiger Reihenfolge mit ihnen die Türen öffnen. Source Code (BASIC) ist auch verfügbar.

HAM 256 Viewer

http://www.worldofspectrum.org/infoseekid.cgi?id=0027262

Andrew Owen hat ein Utility vorgestellt mit dem man am Spectrum mit ULA+ farbenfrohere (mit 256 statt 64 Farben gleichzeitig, 32 pro Zeile) Bilder sich

anschauen kann. Da diese Hardware aber noch nicht in Produktion ist, kann ich das Programm noch nicht testen.



Source Code ist dabei.

Tödliche Straßen

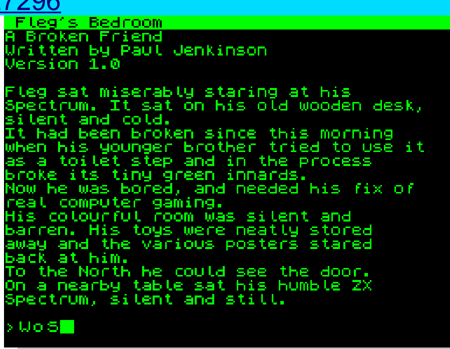
http://www.worldofspectrum.org/infoseekid.cgi?id=0027253



Rafal Miazga hat endlich die Fortsetzung für „Skyscraper of Doom“ fertiggestellt: „Streets of Doom“. Man bekommt wieder die bewährte gute Rätselkost vorgelegt, aber auch Verbesserungen kommen nicht zu kurz. Eindeutig eine Empfehlung wert.

Freundschaftsbruch

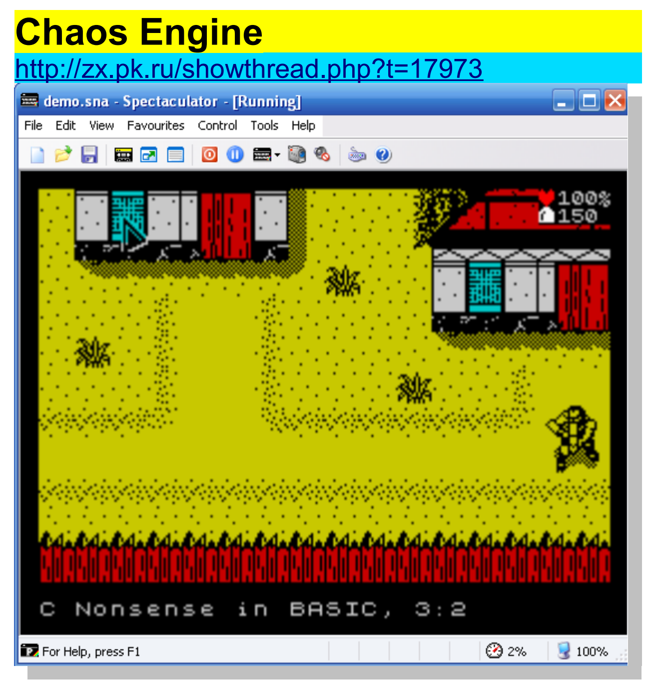
http://www.worldofspectrum.org/infoseekid.cgi?id=0027296



A broken friend ist ein neues Textadventure. Das ungewöhnliche daran ist, dass es nicht mit einem normalen Authoring Tool wie PAW oder GAC, sondern mit Boriels ZX BASIC Compiler erstellt wurde, was erneut seine Vielseitigkeit untermauert.

Chaos Engine

http://zx.pk.ru/showthread.php?t=17973



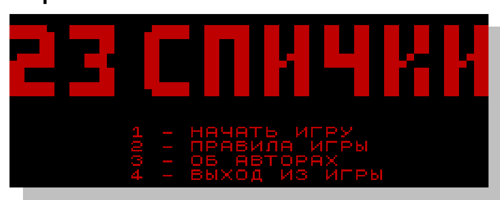
SaNches arbeitet an einem RPG/Action Spiel und hat ein Demo veröffentlicht mit zu 80% fertiger Engine. Diese wird mit K-Mouse oder Tasten gesteuert und es wird Grafik aus Chaos Engine benutzt. Turbo Modus und 128Kb wird vorausgesetzt.
SaNches sucht noch nach Grafikern und Musikern die ihn bei dem Projekt unterstützen wollen.

23

http://www.worldofspectrum.org/infoseekid.cgi?id=0027298

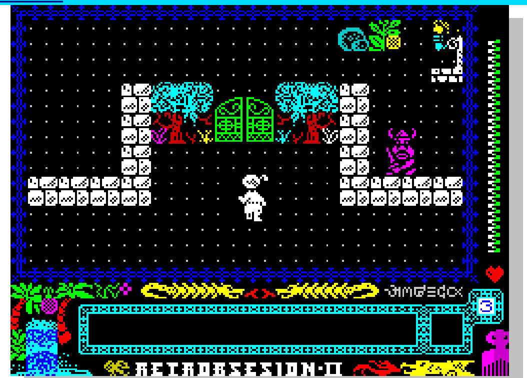
Fishbone hat mit 23 Spichky einen neuen Puzzler geschrieben. Mein Russisch ist zwar nicht ideal, aber ich glaube, dass es so eine Art NIM ist. Man muss also immer eine gewisse Menge an Hölzern entfernen von den 23, bis am

Ende der Gegner gezwungen ist das letzte zu nehmen. Wie bereits gesagt, ist das Spiel auf Russisch.



Retrobsession II

http://www.boriel.com/forum/gallery/retrobsesion-ii-t730.html



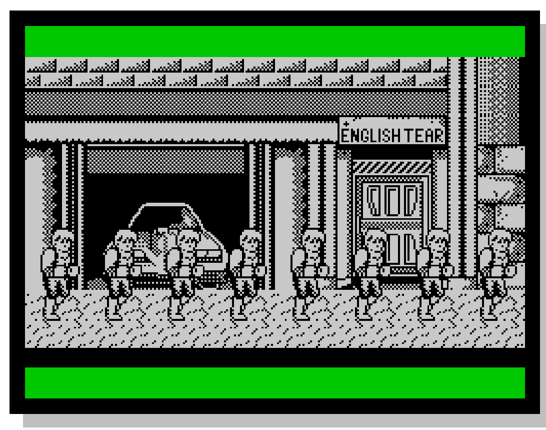
JBGV setzte seinen Vorjahreshit fort. Mit Boriels ZX BASIC Compiler schuf er ein weiteres geniales Spiel welches ich wirklich sehr gerne ausprobiert habe. Es gibt auch ein Memory-Minigame in dem Adventure zu entdecken. Der Download beinhaltet englische und spanische Fassung. Und falls es noch jemand nicht bemerkt hat: Wie beim ersten Retrobsession sind auch hier alle Grafiken aus anderen Spielen geklaut.

Doppelter Drache

http://www.worldofspectrum.org/forums/showthread.php?t=37894

Das Double Dragon Remake (nein, nicht für PC!) nimmt (extrem) langsam seine Formen an. Jim Bagley hat offenbar das erste Technikdemo veröffentlicht, bei dem ACHT große maskierte Sprites flüssig animiert werden.

Wenn das in dem Tempo weiter geht, kommt das Spiel rechtzeitig zum hundertsten Jubiläum der Geburt des ZX Spectrum.



Arkanoid Returns

http://www.worldofspectrum.org/forums/showthread.php?t=37889



Baron Ashler von Kabuto Factory hat einen Karkout Klon angekündigt und ein Demo freigegeben. A.R.C.O.S. ist ein BASIC Kompilat, das ein Mix ist zwischen Krakout und R-Type.

Viel wert wurde vor allem auf die Grafik gelegt, so sieht man u.a. Paralaxscrolling.

Harlequin

http://www.worldofspectrum.org/forums/showthread.php?t=37746
http://trastero.speccy.org/cosas/JL/Harlequin/superfo1.html

Harlequin ist ein Spectrum-Klon der zu 100% kompatibel zu Spectrum 48K ist, auch was das Timing betrifft, allerdings nachgebaut komplett aus normalen TTL-Bausteinen. Optional hat der Rechner einen RGB Ausgang. Es wurden

Sammelbestellungen für eine Selbstbauplatine gesammelt, die zu 100% in ein normales Gummitasten-Gehäuse passt.



Das Foto zeigt ein Prototyp bei dem der Bus leider um ca. 1 cm verschoben war, aber die neue Version hat den Fehler nicht mehr.
Die Platine kostet 10,-, und die Bauteile ohne RGB Chip etwa 30-40 Euro.



Trotz der TTL Bausteine zieht der Klon nur 120 mA an Strom, und somit weniger als ein richtiger Spectrum. Richtig interessant wird wohl erst ein erweiterter Klon wie Chrome, mit 128K, 7 MHz, +D und anderen Spielereien. Da müsste man ein Board anbieten.

Cyborg Terminator 2 – New Future

http://pouet.net/prod.php?which=58270



Ein Spiel welches nach vielen Jahren (geschrieben 1995) endlich releast wird, ist Cyborg Terminator 2. Zwar ist es sicher nicht das am schönsten animierte Game, aber auf jeden Fall interessant. Die Grafik ist sehr detailiert.

Killer III

http://pouet.net/prod.php?which=58272



Auch bei diesem Spiel aus dem Jahre 1995 fragt man sich wieso der Coder es so lange vor uns versteckt hat. Killer III ist ein ganz guter CABAL-Klon vom selben Coder wie Cyborg Terminator 2.

ZX BASIC Compiler

http://www.boriel.com/wiki/en/index.php/ZX_BASIC:Archive

Boriel hat den ZX BASIC Compiler auf version 1.2.9s801 upgedatet.

Es wurden nur Bugfixes gemacht, u.a. bei Tabellen-Kopierfunktion. Boriel hat allerdings bereits die Arbeit am 2.0 Zweig aufgenommen, die auch andere Architekturen unterstützen soll. Leider ist es dazu nötig die komplette Software umzuschreiben.

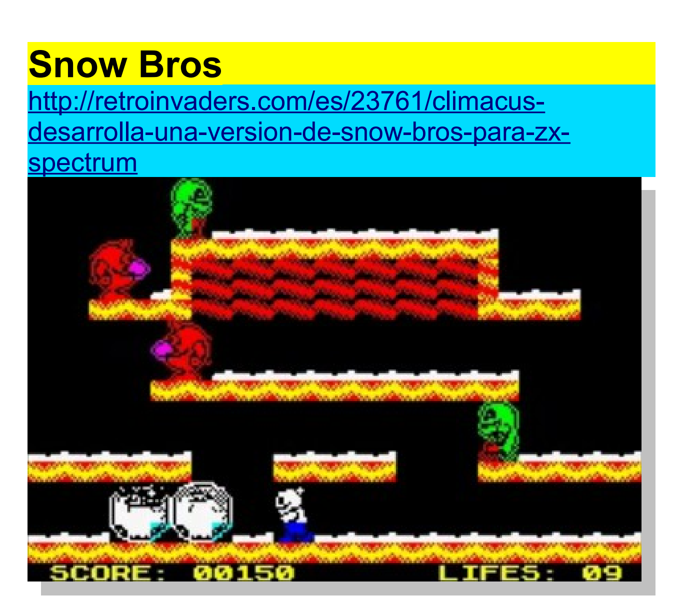
Pip the Pipistrelle

http://www.worldofspectrum.org/forums/showthread.php?t=38132

Am Horizont sieht man, dass ein neues Spiel mit JSW Engine erscheint. Das Besondere daran ist, dass es zehn versteckte Tribute-Screens gibt, z.B. Pitfall oder Nebulus.
Jammajup meinte im WOS-Forum, dass es demnächst erscheinen wird.

Snow Bros

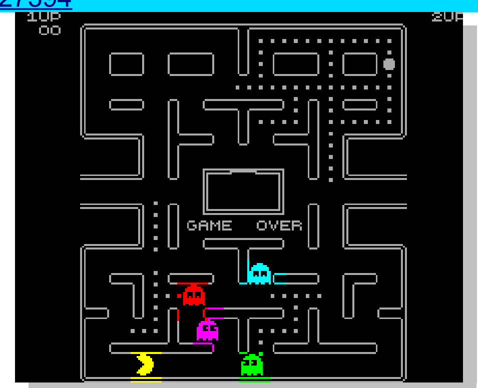
http://retroinvaders.com/es/23761/climacus-desarrolla-una-version-de-snow-bros-para-zx-spectrum



Climacus schreibt nach Retro Invaders sein drittes Spectrum-Spiel, und es sieht nach der Umsetzung des Spielhallenhits „Snow Bros“ aus, welches seinerseits gewisse Ähnlichkeiten mit „Bubble Bobble“ aufweist.

Pac man Technicolor

http://www.worldofspectrum.org/infoseekid.cgi?id=0027394



Simon Owen hat seinen PacMan Emulator für Spectrum+2A/+3 in Farbe fertiggestellt. Best Pacman ever! Mehr muss ich dazu nicht sagen.

SG Software Games

http://www.sg-software.ru/zxspectrum/games



SG Software ist wohl einer der wenigen russischen ZX Programmierern die Software im TAP Format bieten. Nums ist deren aktuellstes Programm, aber ich empfehle wegen der Sprachbarriere, noch andere sich näher anzusehen. Z.B. Waterman, welches entfernt an Pipemania erinnert.

Das DivIDE Plus Teil 3

Ein kleiner Nachteil von MDOS3 ist, dass es beim Einschalten des Spectrum immer neu aktiviert und initialisiert werden muss. Nach dem Umschalten mit *OUT 23,6* muss also zwingend *PRINT USR 0* ausgeführt werden und über das NMI müssen die benötigten Disketten erneut eingelegt werden. Allerdings macht diese Firmware einen sehr professionellen Eindruck und auf dem MD3-Image in der bereits erwähnten Datei "mdos3-images.zip" sind einige interessante Programme zu finden, u.a. ein komfortabler Disk-Manager "Tools 128", der mit dem bekannten "Norton Commander" vergleichbar ist. Außerdem enthalten viele der virtuellen Disketten ein schönes Startmenü, welches man einfach mit *RUN* starten und mit einer Kempston Maus bedienen kann. Eine intensivere Beschäftigung mit MDOS3 lohnt sich auf jeden Fall.

***OUT 23,72 (OUT 23,104)* ... Firmware „ResiDOS 2.25“**

ResiDOS ist mit Abstand die komplexeste Firmware für das DivIDE Plus und so nimmt allein die Beschreibung dieser Firmware gut die Hälfte des DivIDE Plus Handbuches ein. Diese Firmware läuft als einzige vollständig im RAM und benutzt hierfür einen speziellen Modus des DivIDE Plus (dazu später mehr). Auch ist ResiDOS die einzige Firmware für das DivIDE Plus, die lesend und schreibend auf IDE-Datenträger mit FAT-Dateisystem zugreifen kann.

Wenn das DivIDE Plus ausgeliefert wird, ist ResiDOS normalerweise vorinstalliert. Bei meinem Gerät funktionierte ResiDOS leider nicht richtig, außerdem gab es inzwischen eine neue Version und so empfiehlt es sich, ResiDOS komplett neu zu installieren. Dazu besucht man die Homepage von ResiDOS [10] und unter Downloads findet man Dateien für verschiedene Speicher-Interfaces. Hier sucht man die Datei "residos.tap" für das DivIDE Plus [11] heraus und kopiert diese einfach auf eine CF-Karte im FAT-Format.

Nun steckt man diese Speicherkarte in das DivIDE Plus und startet den Spectrum mit gedrückter Leertaste. Wir erinnern uns: Beim Einschalten des DivIDE Plus wird FATware aktiv und durch die Leertaste wird eine neue IDE-Suche ausgelöst. Hier sollte unsere Speicherkarte angezeigt werden und ein beliebiger Tastendruck führt uns in den normalen Spectrum-Bildschirm. Jetzt wird NMI gedrückt und im Disk Browser die Datei "residos.tap" ausgewählt. Dabei landen wir wieder im Spectrum-BASIC und geben *LOAD ""* ein. Eine Sicherheitsabfrage wird mit ENTER bestätigt und der Installer zeigt zunächst die erkannte RAM-Größe von 512kB an. Ein weiterer Druck auf ENTER startet schließlich die Installation. Dies dauert einige Sekunden, dann startet der Spectrum mit ResiDOS neu und zeigt einige Informationen über das System an.

Wenn ResiDOS einmal installiert ist, bleibt es natürlich durch die Batterie-Pufferung auch beim Ausschalten des Spectrum im Speicher und kann beim nächsten Systemstart mit *OUT 23,72*

bzw. *OUT 23,104* wieder aktiviert werden. Dabei bleibt auch der zuletzt verwendete Ordner gespeichert. Wünscht man eine Neuinitialisierung des Systems, dann kann man dies mit *PRINT USR 0* auslösen.

Im Handbuch des DivIDE Plus und auch auf der Download-Seite von ResiDOS wird folgende sehr umständliche Prozedur zum Starten von ResiDOS empfohlen:

POKE 18455,237: POKE 18456,65: POKE 18457,199
RANDOMIZE USR 18455

Das entspricht dem folgenden kurzen Maschinencode-Programm *OUT (C),B* und *RST 0*, wobei die Speicheradressen für die POKEs absichtlich so gewählt wurden. So wird beim Aufruf des Programms mit *RANDOMIZE USR 18455* der Wert 18455 im Doppelregister BC an das Maschinencode-Programm übergeben. Das entspricht dem hexadezimalen Wert 4817H, so dass Register B=48H und C=17H erhält und das wiederum entspricht in dezimaler Schreibweise B=72 und C=23. Damit wird *OUT (C),B* zu *OUT (23),72* und das anschließende *RST 0* ist ein Sprung zur Adresse 0. Um es kurz zu machen: Dieses Maschinencode-Programm tut praktisch das gleiche wie folgende Eingabe in BASIC: *OUT 23,72* und anschließend *PRINT USR 0*.

Im Handbuch des DivIDE Plus wird noch ein weiteres *OUT*-Kommando erwähnt:

OUT 23,104 ... Dieses Kommando sollte laut Handbuch immer dann benutzt werden, wenn man mit Programmen oder Daten im Speicher zu ResiDOS wechselt, um diese mit ResiDOS weiter zu bearbeiten und zu speichern. Grundsätzlich lässt sich ResiDOS immer mit *OUT 23,104* anstatt mit *OUT 23,72* starten - der Unterschied liegt darin, dass der RAM-Bereich, in dem ResiDOS installiert ist, beim Wert *72* schreibgeschützt ist. Warum man hier einen Unterschied macht, ist mir nicht ganz klar. Ich arbeite grundsätzlich nur mit *OUT 23,104* und habe bisher keine Nebenwirkungen feststellen können.

Wie schon anfangs erwähnt, ist ResiDOS sehr komplex. Es kann direkt auf Datenträger mit FAT16-Struktur zugreifen, außerdem auch mit IDEDOS umgehen, welches beim Spectrum +3 mit der IDE-Erweiterung +3e verwendet wird. Die Bezeichnung der Laufwerke erscheint zunächst etwas ungewohnt, so wird bei der Initialisierung des Systems mit *PRINT USR 0* auf dem DivIDE Plus mit angesteckter CF-Karte ein Laufwerk "0A:" erzeugt. Der Buchstabe "A:" entspricht wie bei einem PC dem ersten Laufwerk, die Zahl "0" entspricht einer User Area, die bei IDEDOS Verwendung findet und der älteren Generation vielleicht noch von CP/M bekannt ist. Weiterhin kann ResiDOS natürlich auch mit Pfadnamen umgehen, wobei hier beide Schrägstriche "/" und "\" verwendet werden können.

ResiDOS setzt das erste Laufwerk als aktuelles Laufwerk und alle Kommandos ohne zusätzliche Laufwerksangaben beziehen sich immer auf das aktuelle Laufwerk. Soll ein Kommando z.B. auf Laufwerk B: angewendet werden, dann wird einfach "B:" vor den Dateinamen gesetzt. Die Kommandos sind teilweise

dem MS-DOS sehr ähnlich und werden in ResiDOS durch ein Prozent-Zeichen "%" eingeleitet:

%dir ... zeigt den Inhalt des aktuellen Ordners
CAT ... zeigt den Inhalt des aktuellen Ordners (*CAT* als Token)
%cd "games" ... in den Unterordner "games" wechseln
%cd ".." ... in den darüber liegenden Ordner wechseln
%cd "/" ... in den Wurzelordner wechseln
%md "pics" ... legt den Ordner "pics" an
%rd "demos" ... löscht den Ordner "demos" (muss leer sein)
%era "prog2" ... löscht die Datei "prog2"
%del "bytes" ... löscht die Datei "bytes"
ERASE "data" ... löscht die Datei "data" (*ERASE* als Token)
%cp "dat1","/games/d" ... kopiert die Datei "dat1" aus dem aktuellen Ordner in den Ordner "/games/d"
%cp "/progs/lister","" ... kopiert die Datei "lister" aus dem Ordner "/progs" in den aktuellen Ordner
%ren "test1","game1" ... benennt die Datei "test1" in "game1" um

Es können auch Wildcards (* und ?) verwendet werden, um z.B. mehrere Dateien zu kopieren oder zu löschen.

Programme und Daten (BASIC, CODE und SCREEN$) können direkt mit *SAVE* auf dem FAT-Datenträger gespeichert oder mit LOAD vom Datenträger geladen werden. Dabei werden alle Dateien mit einem zusätzlichen Header versehen, damit sie auch vom +3e System gelesen werden können.

SAVE %"prog1" ... speichert das Programm "prog1" im aktuellen Ordner
SAVE %"bytes" CODE 32768,16384 ... speichert den CODE-Block "bytes" im aktuellen Ordner
LOAD %"prog2" ... lädt das Programm "prog2" aus dem aktuellen Ordner

Es können übrigens auch beliebige Dateien vom PC geladen werden, z.B. Texte. Dazu kann man das Kommando *LOAD %"name"* CODE verwenden und ohne weitere Angaben wird an die Speicherposition 32768 geladen.

Grundsätzlich verwendet ResiDOS bei den Dateinamen die von MS-DOS bekannte 8.3 Schreibweise, anstatt "prog1" sollte also besser "prog1.bas" geschrieben werden. So lassen sich die Dateien sowohl auf dem Spectrum als auch auf einem PC besser zuordnen und verwalten. Die Groß-/Kleinschreibung ist hier nicht von Bedeutung, auf dem Datenträger werden immer Großbuchstaben verwendet und ResiDOS wandelt die Zeichen entsprechend um. Leider unterstützt ResiDOS keine langen Dateinamen, ein Zugriff auf solche Dateien ist aber trotzdem über den entsprechenden Kurznamen (z.B. "progra~1.tap") möglich.

Selbstverständlich kann ResiDOS auch Snapshot-Dateien (*SNA* und *Z80*) laden, dafür gibt es ein spezielles *LOAD*-Kommando:

%snapload "demo1.sna" ... lädt und startet das Programm "demo1.sna"
%snapload "farben.z80" ... lädt und startet das Programm "farben.z80"

ResiDOS kann noch vieles mehr, aber das würde den Rahmen dieses

Beitrages sprengen. Auf eine Funktionalität möchte ich aber dennoch eingehen: ResiDOS kann mit zusätzlichen Modulen erweitert werden und folgende Module sind derzeit verfügbar:

- Channels - ermöglicht das Schreiben und Lesen von Datenfeldern und die komfortable Verwendung von Bildschirmfenstern
- TapeIO - dient zum Lesen und Schreiben von *TAP* und *TZX* Dateien
- TaskMan - aktiviert einen Task-Manager über ein NMI-Menü, ermöglicht die Umschaltung zwischen mehreren Spectrum-Programmen und das Erstellen von Snapshots
- ZX80 Emulator - ermöglicht das Ausführen von *.O* Dateien (nur auf Spectrum 128k)
- ZX81 Emulator - ermöglicht das Ausführen von *.P* Dateien (nur auf Spectrum 128k)

Ein solches Modul (genannt Package) wird immer nach dem gleichen Prinzip installiert, als Beispiel wähle ich hier TapeIO. Zunächst muss man das gewünschte Package von der ResiDOS-Homepage laden [12] und anschließend auf eine CF-Karte kopieren. Nach dem Einschalten des Spectrum und dem Umschalten auf ResiDOS mit *OUT 23,104* wechselt man in den Ordner, in dem die Datei "tapeio.pkg" abgelegt ist. Weiter geht es mit folgenden Kommandos:

CLEAR 32767 ... RAMTOP setzen
%install "tapeio.pkg" ... Package installieren

Anschließend kann man sofort mit dem neuen Modul arbeiten und z.B. eine *TAP*-Datei laden:

%tapein "/games/j/jumping.tap" ... die *TAP*-Datei "jumping.tap" als virtuelles Band zum Lesen einlegen
LOAD "" ... Laden vom virtuellen Band

TapeIO kann auch virtuelle Bänder schreiben, man kann den Inhalt eines Bandes auflisten, zu beliebigen Positionen springen und mehr. Für weitere Informationen über TapeIO und generell über ResiDOS verweise ich auf das DivIDE Plus Handbuch [3] und die Homepage von ResiDOS [10]. Dort findet man die komplette Kommando-Referenz und die Beschreibung weiterer Möglichkeiten.

3.2 Installierbare Firmware

Grundsätzlich besteht beim DivIDE Plus die gleiche Möglichkeit wie beim DivIDE, den Flash-Speicher mit einer neuen Firmware zu beschreiben. Allerdings können die Installer, die für das kleine DivIDE gedacht sind, hier nicht verwendet werden. Das liegt zum einen daran, dass der Flash-Speicher einen anderen Programmier-Algorithmus erfordert und zum anderen an der zusätzlichen Bankumschaltung, die von den Installern nicht berücksichtigt wird. Überhaupt scheint es derzeit gar keine Firmware zu geben, die direkt im Flash-Speicher des DivIDE Plus installiert werden kann. Man fragt sich natürlich, wie die Erbauer des DivIDE Plus die Firmware in den Flash-Speicher bekommen haben und die Antwort ist ganz einfach: mit einem

Programmiergerät.

Es gibt allerdings eine Ausnahme: die Software "Firmware Selector" lässt sich zusätzlich fest auf dem DivIDE Plus installieren. Hierbei handelt es sich um ein Startmenü, welches eine bequeme Auswahl der gewünschten Firmware ermöglicht. Hier kann auch eingestellt werden, ob der Spectrum im 48k- oder 128k-Modus starten soll. Diese Auswahl wird gespeichert und fortan startet das DivIDE Plus immer direkt die gewählte Firmware. Um wieder das Startmenü zu erreichen, muss beim Systemstart die Taste "*1*" gehalten werden. Die manuelle Auswahl einer Firmware über *OUT*-Kommandos bleibt natürlich erhalten. Dabei muss jedoch beachtet werden, dass FATware auf eine andere Speicherbank verschoben wurde und nun mit *OUT 23,28* aktiviert wird.

Der Firmware Selector ist im Prinzip eine schöne Sache, die allerdings ihren Preis hat und bei RWAP Services £5 kostet [2]. Auch hat die Praxis gezeigt, dass man bei häufigem Firmware-Wechsel mit dem *OUT*-Kommando schneller ans Ziel kommt. Außerdem führt der Firmware Selector immer ein *USR 0* durch, was bei einigen Firmwares ungewollt zu einer Neuinitialisierung führt, d.h. einige Einstellungen gehen verloren (z.B. ein gewählter Ordner). Aus diesem Grund habe ich den Firmware Selector von meinem DivIDE Plus wieder entfernt. Für diesen Zweck wird ein De-Installer mitgeliefert, der dann auch FATware wieder an die richtige Stelle bringt.

Neben dem Firmware Selector als derzeit einzige fest installierbare Software gibt es noch weitere Firmware, die man allerdings nur temporär im RAM installieren kann. Das liegt daran, dass der MAPRAM-Modus verwendet wird, der nur so lange aktiv bleibt, wie der Spectrum eingeschaltet ist. Außerdem wird immer die Speicherbank 0 verwendet und hier wird das RAM auch von FATware benutzt.

Man kann aber mit diesem kleinen Nachteil leben, denn die Installation einer solchen Firmware ist meist in wenigen Sekunden erledigt. Für diesen Zweck kann man einfach FATware verwenden: Die *TAP*-Datei mit der zu installierenden Firmware kommt ganz normal auf ein CF-Karte, wird dann im Disk Browser ausgewählt und anschließend mit *LOAD ""* gestartet. Eventuelle Hinweise auf den zu entfernenden Jumper für den EEPROM-Schreibschutz kann man ignorieren. Das DivIDE Plus wechselt dann in den MAPRAM-Modus und die neue Firmware wird im RAM installiert.

Firmware "TBIOS 1.4i"

Diese Firmware ist gut zum Testen der DivIDE-Hardware geeignet. Allerdings ist diese Software für das kleine DivIDE gedacht und so wird auf einem DivIDE Plus nur der Speicher-Bereich 0 getestet. Auf der Velesoft-Homepage [13] wird bei TBIOS auf einige weitere Funktionen hingewiesen, diese konnte ich aber mangels genauer Informationen nicht wirklich sinnvoll anwenden.

Firmware "FATware 128"

FATware 128 ist eine Anpassung der bekannten FATware 0.12 an den

Spectrum 128 und wird ebenfalls bei Velesoft [13] zum Download angeboten. Auch diese Firmware ist für das klassische DivIDE gedacht und beim DivIDE Plus nicht wirklich notwendig. Sie lässt sich aber dennoch installieren und ausführen.

Firmware "ESXDOS 0.73 Beta"

ESXDOS ist eine relativ neue Firmware (obwohl schon recht lange dran gearbeitet wird) und ebenfalls für das kleine DivIDE gedacht. Auf diesem lässt sie sich auch problemlos installieren und starten. Leider läuft sie nicht auf dem DivIDE Plus und das ist sehr schade, da sie als einzige Firmware mit FAT32 umgehen kann (zumindest lesend); auf FAT16-Medien ist sowohl Lesen als auch Schreiben möglich.

Firmware "BS-DOS 309"

Diese Firmware stammt vom MB02+ Interface und wurde an das DivIDE angepasst. BS-DOS verwendet kein bekanntes Dateisystem und greift direkt auf die Sektoren des Datenträgers zu. Es arbeitet mit virtuellen Disketten, die eine Größe von bis zu 2MB haben können. Insgesamt werden bis zu 255 Disketten verwaltet.

Auf der BS-DOS Homepage von Velesoft findet man den Installer für BS-DOS [14]. Nach der Installation wird ein passend eingerichteter Datenträger erwartet, damit man überhaupt arbeiten kann. Das ist allerdings ein Problem, denn startet man die Firmware von einem FAT-Datenträger, dann wird natürlich anschließend kein passender Datenträger vorgefunden. Legt man vorher den entsprechenden Datenträger ein, dann kann man die Firmware nicht davon installieren. Ein Wechsel des Datenträgers im laufenden Betrieb ist nicht empfehlenswert und so gibt es nur eine Alternative: Man muss zum Installieren der Firmware entweder die gute alte Kassette oder einen zweiten Datenträger benutzen, den man z.B. über einen Doppel-CF-Adapter anschließen kann.

Auch das Einrichten eines Datenträgers ist nicht so einfach, da man zunächst eine spezielle Partition erstellen und anschließend virtuelle Disketten formatieren muss. Kurz gesagt - ein Neuanfang ist nicht unbedingt empfehlenswert. BS-DOS ist aber trotzdem interessant, da es einen direkten Datenaustausch mit dem recht weit verbreiteten MB02+ ermöglicht. Leider habe ich selbst kein MB02 und auch keinen Besitzer eines solchen in der Nähe. Deshalb werden weitere Experimente auf eines der nächsten Usertreffen verschoben.

4. Das OUT 23 Kommando

Der Port 23 dient hauptsächlich zur Verwaltung des Speichers des DivIDE Plus, bietet aber noch mehr Funktionen. Es ist ein reiner Ausgabe-Port, d.h. es können nur Werte mittels *OUT 23,x* ausgegeben werden. Ein Lesen des Ports mit *PRINT IN 23* ist zwar möglich, liefert aber immer 0 als Ergebnis. Die Entwickler haben sich einige Gedanken über die Speicherverwaltung gemacht und es ist durchaus interessant, sich mit diesen Möglichkeiten zu beschäftigen.
Da einzelne Bits des Ports 23 eine

bestimmte Funktion haben, sollte man den auszugebenden Wert immer binär betrachten. Eine besondere Bedeutung haben die beiden höchstwertigen Bits 7 und 6: Sie bestimmen den Modus des DivIDE Plus und je nach Modus haben die restlichen Bits eine unterschiedliche Bedeutung.

DivIDE-Modus (Bit 7 = 0, Bit 6 = 0)

In diesem Modus wird der gesamte Speicher des DivIDE Plus in 16 Bereiche mit jeweils 32kB ROM und 32kB RAM unterteilt und in jedem dieser Bereiche kann eine Firmware installiert werden. Über das folgendermaßen aufgebaute *OUT 23* Kommando kann ein bestimmter Bereich und somit eine bestimmte Firmware ausgewählt werden:

Bit 7 ... 0 \ DivIDE-Modus
Bit 6 ... 0 / aktivieren
Bit 5 ... 0
Bit 4 ... \ Auswahl des Bereiches
Bit 3 ... \ Auswahl des Bereiches
Bit 2 ... / Auswahl des Bereiches
Bit 1 ... / Auswahl des Bereiches
Bit 0 ... 0

Die Bits 4 bis 1 bestimmen in binärer Form den ausgewählten Bereich. Innerhalb eines solchen Bereiches verhält sich das DivIDE Plus wie ein klassisches DivIDE und arbeitet im so genannten Kompatibilitäts-Modus. Dabei wird von den zur Verfügung stehenden 32kB ROM ein fester 8kB großer Block im Adressbereich 0-8191 (0000-1FFF) eingeblendet, während die 32kB RAM nochmals durch eine Bank-Umschaltung in 4 jeweils 8kB große Blöcke unterteilt sind. Einer dieser Blöcke wird dann im Adressbereich 8192-16383 (2000-3FFF) eingeblendet. Dafür wird der I/O-Port 227 (E3) verwendet, welcher übrigens durch die Firmware selbst verwaltet wird, sofern sie von der Bank-Umschaltung Gebrauch macht. All das ist übrigens auch beim klassischen DivIDE so realisiert.

Dieser Modus wird von folgenden Firmwares benutzt:

OUT 23,BIN 00000000 (OUT 23,0) ... FATware auf Bereich 0 (Firmware Selector ist nicht installiert)
OUT 23,BIN 00011100 (OUT 23,28) ... FATware auf Bereich 14 (Firmware Selector ist installiert)
OUT 23,BIN 00000010 (OUT 23,2) ... DEMFIR auf Bereich 1
OUT 23,BIN 00000100 (OUT 23,4) ... +DivIDE auf Bereich 2
OUT 23,BIN 00000110 (OUT 23,6) ... MDOS3 auf Bereich 3

Eine besondere Variante des DivIDE-Modus ist der MAPRAM-Modus. Hier wird eine spezielle Speicherkonfiguration aktiv, die anstatt des 8kB ROM im Adressbereich 0-8191 (0000-1FFF) eine RAM-Bank einblendet. Daraus ergibt sich im unteren Speicherbereich des Spectrum ein durchgehender 16kB großer RAM-Block. Das ermöglicht die Installation einer Firmware im RAM, wobei der ROM-Inhalt nicht verändert wird.

Um eine Firmware im MAPRAM Modus zu installieren, sind keine besonderen Vorarbeiten notwendig, im Gegenteil: Wenn bei der Installation der Hinweis zum Entfernen des Schreibschutz-Jumpers erscheint, wird dieser einfach

ignoriert und ENTER gedrückt. Dabei aktiviert der Installer über den bereits erwähnten Port 227 den MAPRAM Modus und die Firmware wird im RAM installiert.

Übrigens: die Firmwares DEMFIR und MDOS3 schalten nach dem Start auch in den MAPRAM Modus um. Der aktivierte MAPRAM Modus ist am Leuchten der gelben LED zu erkennen. Er kann nur durch ein DivIDE-Reset mittels *OUT 23,192* (Erklärung folgt noch) oder durch Abschalten der Stromversorgung wieder verlassen werden.

Im DivIDE-Handbuch wird noch ein weiterer Sonderfall erwähnt: Der DivIDE Plus 128k Modus. Allerdings konnte ich noch nicht herausfinden, wofür man diesen Modus verwenden kann. Der 128k Modus wird durch folgende Bit-Belegung aktiviert:

Bit 7 ... 0 \ DivIDE Plus
Bit 6 ... 0 | 128k Modus
Bit 5 ... 1 / aktivieren
Bit 4 ... 0
Bit 3 ... 0
Bit 2 ... 0
Bit 1 ... 0
Bit 0 ... 0
entspricht: *OUT 23,32*

RAM-Modus (Bit 7 = 0, Bit 6 = 1)

In diesem Modus verwendet das DivIDE Plus nur das RAM und unterteilt dieses in 32 Bereiche mit jeweils 16kB. Diese 16kB werden komplett im Adressbereich 0-16383 (0000-3FFF) eingeblendet, so dass sich für den Spectrum ein durchgehendes RAM im gesamten Adressbereich ergibt. Über das folgendermaßen aufgebaute *OUT 23* Kommando kann ein bestimmter Bereich ausgewählt werden:

Bit 7 ... 0 \ RAM-Modus
Bit 6 ... 1 / aktivieren
Bit 5 ... Schreibzugriff sperren oder erlauben
Bit 4 ... \ Auswahl des Bereiches
Bit 3 ... \ Auswahl des Bereiches
Bit 2 ... | Auswahl des Bereiches
Bit 1 ... / Auswahl des Bereiches
Bit 0 ... / Auswahl des Bereiches

Über die Bits 4 bis 0 wird der gewünschte RAM-Bereich ausgewählt. Mit Bit 5 kann der Schreibzugriff auf den Adressbereich 0-16383 (0000-3FFF) gesteuert werden, eine 0 entspricht einem Schreibschutz und eine 1 erlaubt den Schreibzugriff. Die einzige Firmware, die den RAM-Modus benutzt, ist ResiDOS:

OUT 23,BIN 01001000 (oder *OUT 23,72*) ... ResiDOS auf Bereich 8 mit Schreibschutz aktivieren
OUT 23,BIN 01101000 (oder *OUT 23,104*) ... ResiDOS auf Bereich 8 ohne Schreibschutz aktivieren

ROM-Modus (Bit 7 = 1, Bit 6 = 0)

In diesem Modus verwendet das DivIDE Plus nur das ROM und unterteilt dieses in 32 Bereiche mit jeweils 16kB. Diese 16kB werden komplett im Adressbereich 0-16383 (0000-3FFF) eingeblendet. Folgendes *OUT 23* Kommando ermöglicht die Auswahl eines bestimmten Bereiches:

Bit 7 ... 1 \ ROM-Modus

Bit 6 ... 0 / aktivieren
Bit 5 ... Schreibzugriff sperren oder erlauben
Bit 4 ... \ Auswahl des Bereiches
Bit 3 ... \ Auswahl des Bereiches
Bit 2 ... | Auswahl des Bereiches
Bit 1 ... / Auswahl des Bereiches
Bit 0 ... / Auswahl des Bereiches

Auch hier bestimmen die Bits 4 bis 0 die Auswahl des gewünschte Bereiches. Bit 5 dient hier ebenfalls zur Steuerung des Schreibzugriffs auf den Adressbereich 0-16383 (0000-3FFF), wobei eine 0 den Schreibzugriff sperrt und eine 1 den Zugriff erlaubt. Ein versehentliches Überschreiben des FlashROM ist jedoch nicht möglich, da hierfür ein spezieller Algorithmus erforderlich ist. Zurzeit ist mir keine Firmware bekannt, die den ROM-Modus benutzt.

Reset-Modus (Bit 7 = 1, Bit 6 = 1)

Mit diesem Modus können alle internen Speicherbank-Register des DivIDE Plus zurückgesetzt werden:

Bit 7 ... 1 \ Reset-Modus
Bit 6 ... 1 / aktivieren
Bit 5 ... 0
Bit 4 ... 0
Bit 3 ... 0
Bit 2 ... 0
Bit 1 ... 0
Bit 0 ... 0
entspricht: *OUT 23,192*

Nach dem Absenden dieses Kommandos sollte man auch beim Spectrum ein Reset ausführen. Der Reset-Modus ist hauptsächlich dafür gedacht, den MAPRAM-Modus des DivIDE Plus wieder zu deaktivieren. Beim kleinen DivIDE musste dafür immer die Stromversorgung ausgeschaltet werden und auch beim DivIDE Plus kann das notwendig sein, wenn sich das System im MAPRAM-Modus nicht mehr bedienen lässt.

5. Sonstiges

Kompatibilität zum klassischen DivIDE: Ich bin der Meinung, die Entwickler haben sich große Mühe gegeben, um das DivIDE Plus bestmöglich kompatibel zu seinem Vorgänger zu machen. Das bedeutet, dass alle Programme, die auf dem DivIDE laufen, auch auf dem DivIDE Plus laufen sollten und mir ist kein Programm bekannt, welches hier eine Ausnahme macht. Aber es muss Unterschiede geben, was man an der Firmware ESXDOS erkennen kann, die definitiv nicht auf dem DivIDE Plus funktioniert.

Hinweis für Probleme mit CF-Karten, die von FATware zwar erkannt, jedoch nicht verwendet werden können (Meldung "0 volume(s) found"): Beim Präparieren einer CF-Karte für die Firmware BS-DOS bin ich auf das Programmpaket "zx-ide-utils.zip" gestoßen, welches man auf der DivIDE-Seite von Velesoft [13] laden kann. Dieses enthält einen einfachen Partitions-Manager namens "ZX-FDISK", der eine ähnliche Funktionalität wie das von MS-DOS bekannte FDISK besitzt. Laut Velesoft soll es damit möglich sein, eine problematische CF-Karte wieder zur Zusammenarbeit mit FATware zu bewegen. Dazu muss man mit ZX-FDISK einen neuen MBR mit Partitionstabelle erstellen, eine neue

primäre Partition anlegen und anschließend die CF-Karte auf einem PC formatieren.

Ich hatte leider keine problematische CF-Karte zur Verfügung und war auch nicht gewillt, eine Karte für so ein Experiment zu opfern. Deshalb konnte ich nicht prüfen, ob das auch wirklich so funktioniert. Man hat auch nicht erwähnt, wie man ZX-FDISK laden soll. Man kann zwar ZX-FDISK auf die CF-Karte kopieren, aber da FATware diese Karte nicht akzeptiert, scheidet FATware zum Laden aus. Aber man könnte es z.B. mit ResiDOS versuchen.

ZX-FDISK habe ich aber für ein anderes Experiment genutzt. Der Plan war, mehr als eine Partition auf einer CF-Karte anzulegen und so eine universelle CF-Karte zu schaffen, die man im besten Fall mit allen Firmwares verwenden kann. Um es kurz zu machen: man kann zwar mehrere Partitionen anlegen, aber es wird immer nur die erste verwendet und das sowohl von einem PC als auch von den verschiedenen Firmwares. An dieser Stelle habe ich den Plan aufgegeben und jede Firmware bekommt nun eine eigene CF-Karte. Lediglich +DivIDE lässt sich relativ gut auf einem FAT-Medium integrieren und so lassen sich immerhin FATware, +DivIDE und ResiDOS mit der gleichen CF-Karte nutzen.

Und das obligatorische Schlusswort: Beim Schreiben dieses Textes habe ich viel an meinem Spectrum-System gesessen und auch wirklich alles ausprobiert (bis auf die eben genannte Ausnahme ZX-FDISK). Dabei konnten viele neue Erkenntnisse gewonnen werden, die dann auch gleich wieder in diese Beschreibung eingeflossen sind. So hat die Arbeit an diesem Text einige Wochen in Anspruch genommen und ich hoffe, dass anderen DivIDE Plus Nutzer mit diesen Informationen etwas anfangen können. Vielleicht motiviert es ja auch den einen oder anderen zur Anschaffung eines DivIDE Plus, auch wenn es nicht so einfach zu bekommen ist.

Ansonsten gilt: Fragen, Lob und Kritik bitte an s.huehn@soemtron.de senden :-)

6. Links

[1]
http://www.rwapsoftware.co.uk/ ... Homepage der Firma "RWAP Software"

[2]
http://www.rwapsoftware.co.uk/spectrum/spectrum_storage.html ... Auf dieser Seite kann das DivIDE Plus bestellt werden (leider nur selten verfügbar).

[3]
http://www.rwapsoftware.co.uk/spectrum/divide_plus_manual.doc ... Hier kann das Handbuch zum DivIDE Plus geladen werden (nur in englischer Sprache).

[4]
http://demfir.sourceforge.net/ ... DEMFIR-Homepage, hier muss das Paket "demfir-d0.7b.tar.gz" geladen und die Datei "demfird07b.iso" zur vollständigen Installation von DEMFIR extrahiert werden.

[5]
http://s-huehn.de/spectrum/ ... Der Link "Software" führt zum +divide Manager.

[6]
http://www.worldofspectrum.org/documentation.html#manuals ... Hier kann neben vielen anderen Dokumenten das MGT Plus-D Manual geladen werden.

[7]
http://www.biehold.nl/rudy/+divide/index.htm ... +DivIDE-Homepage

[8]
http://velesoft.speccy.cz/zx/divide/divide-mdos3.htm ... Homepage von Velesoft mit Infos über MDOS3.

[9]
http://velesoft.speccy.cz/zx/divide/systems/mdos3/mdos3-images.zip ... MD3-Image mit einigen virtuellen Disketten für MDOS3.

[10]
http://www.worldofspectrum.org/residos/ ... ResiDOS-Homepage

[11]
http://www.worldofspectrum.org/residos/files/divideplus/residos.tap ... Installationsdatei für ResiDOS

[12]
http://www.worldofspectrum.org/residos/files/tapeio.pkg ... TapeIO-Package für ResiDOS

[13]
http://velesoft.speccy.cz/zx/divide/divide-soft.htm ... Homepage von Velesoft für das DivIDE

[14]
http://velesoft.speccy.cz/zx/divide/divide-bs-dos.htm ... Homepage von Velesoft für BS-DOS

Scott Falk Hühn

HOW to Use Inline Assembly

Guide to inline assembly using Boriel's ZX compiler to create zx spectrum code.

© 2012 Britlion - Thanks to all the Wossers, and all the Boriel forum folks (especially Boriel!) who make everything I learn possible. Special shoutout to Gigatron for allowing me to publish this.

Going from BASIC to machine code can be tricky for beginners. Boriels ZX BASIC does make it easier, however, in my opinion. You can plan bits of the logic in basic, and then replace them with pure machine code modules as you go along - and because the compiler outputs pretty fast machine code to begin with, you only really need to hand-assemble any speed critical parts anyway. Regardless, it's a great way to learn machine code on the Z80 - you don't need to know how to write a whole program to give it a try.

What this tutorial is going to discuss is how to write a function in BASIC, and replace it with a pure machine code version instead, plugged straight in in place of the ZX BASIC version.

Do remember that the zx basic version is already compiled, so is already hundreds of times faster than Sinclair BASIC already. This tutorial is more about showing how the compiler lets you replace parts of the code with inline assembly seamlessly.

Let's create a function that games programmers have been known to use. A function that takes a byte representing something graphical and mirrors it left to right, so the bits are the other way around.

So BIN 11010000 (208 in decimal)

becomes the equivalent of BIN 00001011 (11 in decimal).

Here's the BASIC version, with lots of comments to help you through it.

```
FUNCTION mirror (dowedoit as uByte, number as uByte) AS uByte

REM I want this function to only mirror if told to. So if dowedoit=0, that means we don't mirror the bits.

 IF dowedoit=0 THEN RETURN number

 END IF


 DIM loopcount AS uByte : REM this counts our 8 bit loop.

 DIM potentialOutput AS uByte : REM this is the variable we need to hold our part built output.


 REM The algorithm here isn't too difficult to follow, if you think of "number" as 8 bits, instead of a numeric value.

 REM This is wise, since we're going to use this function to mirror pixels for the screen.

 REM How it works is that it takes the bits input, and checks the rightmost column

 REM If there's a 1 there, it puts it in the right hand side of the output, and the input rolls right

 REM and the output rolls left.

 REM The result of this is the same bits in the output as were in the input, but opposite way around.

 REM Input bits 01234567 --> output bits 76543210


 REM this lets us store graphics as facing left, but be flipped for facing right if we need that, for example.


 FOR loopcount=0 TO 7

  LET potentialOutput=potentialOutput*2

  REM this in binary shifts all the bits one to the left, and adds a zero on the end.

  REM It's the same thing in decimal - if you multiply by 10, all the columns move one to the left. Since binary is

  REM a base 2 system, and decimal is base 10, then *2 in binary behaves a lot like *10 in decimal.

  IF number BAND 1 = 1 THEN LET potentialOutput=potentialOutput+1

  END IF


  REM "BAND" is the binary AND function. In this case, we're looking at the rightmost bit, and seeing if it's 1.

  REM If it's 1, we put a 1 in the right most column of the output too.


  LET number=number/2

  REM Just like above, we're rolling the bits right with a divide by 2. Here, the last bit is lost, since in integer maths 1/2=0.

 REM We don't have a binary point for fractions.

 NEXT loopcount


 RETURN potentialOutput

 REM Return sends back the output from the function

END FUNCTION

CLS

REM Here’s how you use it:

PRINT 0,mirror(1,0)

PRINT 1,mirror(1,1)

PRINT "(not mirrored) ";1,mirror(0,1)

PRINT 2,mirror(1,2)

PRINT 3,mirror(1,3)

PRINT 4,mirror(1,4)

PRINT 8,mirror(1,8)

PRINT 16,mirror(1,16)
```

```
PRINT 32,mirror(1,32)
PRINT 64,mirror(1,64)
PRINT 127,mirror(1,127)
PRINT 255,mirror(1,255)
PRINT
PRINT 208,mirror(1,208)
```

```
0                 0
1                 128
1 (unmirrored)    1
2                 64
3                 192
4                 32
8                 16
16                8
32                4
64                2
127               254
255               255

208               11

0 OK, 30:1
```

Boriel's BASIC is far more flexible than Sinclair BASIC in several ways. Firstly line numbers are not required (though they are allowed). You can be more flexible with variable names - here I've tried to use long descriptive ones so that the variable is what programmers call "self documenting" - that is, the name makes it obvious what it does. You should try to be descriptive in your variable names. They don't slow the compiled code down, and don't cost extra memory.

let score=score+10 is far easier to understand later on than

let s=s+10

Sinclair BASIC had very limited memory, so such terseness was often a good idea. With a cross compiler like ZX BASIC, you don't have that limitation. Describe away. The same goes for REM statements - there's no need to scrimp!

One thing that does differ though, that's very important:

You must have an END IF closing off an IF statement. This is the one thing you MUST ALWAYS change when converting Sinclair BASIC for the compiler. There are other differences - some statements aren't (completely) supported, for example - but this is the biggest one to be aware of.

You can see this used above.

Of course, IF is more flexible as a result - for example -

```
IF score=0 THEN PRINT "No Score"
ELSEIF
 score > 10000 THEN PRINT "Amazing Score!"
ELSE
 PRINT score
END IF
```

Hopefully this is self explanatory - if the score is 0, then it prints the words "No Score". If you have a really high score, you get the words "Amazing Score!" and if you have something in between (or for that matter negative), you get the value of the variable score printed.

The other thing that's different is use of variable types. Here we have defined things like loopcounter as a uByte - that is an unsigned byte. It can therefore only take the values 0-255 as it has to fit in one byte of memory. While this limits the values that can be used, it massively increases the processing speed of using it. If we add numbers to it, the resultant code is much simpler - sinclair basic has to do loopcounter=loopcounter+1 using the same code that could handle -2.4+345343.3444, which is clearly a

much more complex case!

So, let’s test what this looks like shall we?

How about this program:

```
DIM n AS uByte
DIM m AS uInteger
CLS
REM Let's print some characters on the screen.
FOR n=32 TO 127
PRINT CHR$(n); CHR$(n);
NEXT n
REM Same thing, in the middle third of the screen.
PRINT AT 8,0;
FOR n=32 TO 127
PRINT CHR$(n); CHR$(n);
NEXT n
PAUSE 0
FOR m=18432 TO 20480 : REM This takes the screen bytes in the middle third
POKE m,mirror(1,PEEK m) : REM peeks them, and pokes them back, mirrored.
NEXT m
```

As you can see, each character in the middle of the screen got mirrored.

If you compile and run this, you’ll need to include the mirror function from earlier. You’ll also see it working, which is quite fascinating - even as compiled code it’s slow enough to take a couple of seconds to run. A similar basic program would probably take minutes to work, however.

But say we think that’s still not fast enough? Can we go faster?

How about changing the mirror function into assembly language/machine code directly? Hand coded assembly is (usually) faster than compiled code - humans are better at the job still.

I’m not going to teach you machine code, here. There are lots of good books and articles on it, especially Toni Baker’s work. What I’ll discuss here is how to convert using Boriel’s compiler.

Let’s convert the mirror function then:

The first thing we’ll do is change the function to a fastcall function. This means that it does less setup than a standard function, and we can “RET” straight out of it, if we want. We also have to clean up the stack ourselves, while making registers fit the promised return type value we set up - in this case we’re returning a uByte, which means our A register must contain our return value.

You don’t have to use a fastcall function to use inline machine code assembly, but it’s often useful to do so for simple functions - especially single parameter functions. I’ve deliberately made this a two parameter function, so you can see how to do it with those.

Fastcall functions are also handed the parameters on the stack, but the first

one comes in on the registers, which is handy.

```
FUNCTION FASTCALL mirror (dowedoit AS uByte, number AS uByte) AS uByte

 REM This function is now machine code, so the parameters arrive on the machine code stack.

 ASM

 POP HL ;' pull our return address from the stack - it's always the first "parameter" because when the main code uses

 ;' "CALL", the return address is stacked.

 ;' we have to get that out of the way first.

 ;' "dowedoit" is a byte, so arrives in the A register. Fastcall functions send the first parameter in the registers.

 POP BC ;' pop our second parameter (Number) into b (it also loads another byte we don't care about into c.

 ;' Since number is a uByte, it fits in an 8 bit register.)

 ;' so now our stack has no parameters at all on it - not even a return address (which is in HL).

 PUSH HL ;' put our return address back on the stack. It's used by ret at the end of this function.

 ;' Note that if this was a single parameter, you wouldn't need to mess with the stack at all.

 ;' A fastcall function gets its first parameter in the registers.


 AND A ;' this tests the A register (without changing it), setting the flags register so we can act on it.

 LD A,B ;' Put "number" that was in B into the A register.


 RET Z ;' return if our flags say that A was at zero when we tested it - that is "dowedoit" was zero.

 ;' Since A now holds "number", we're returning it untouched.

 ;' ret also pops the address off the stack, meaning we get back with a clean stack.


 ;' So if we have to mirror what's in the A register, we get to here. We didn't return.


 LD B,8 ;' Let's use B as a loop counter. It counts down, here.

 LD C,A ;' let's put our original number in c, because we need the A register for output.


 XOR A ;' This (exclusive or) zeroes out the A register. It's shorter and faster than LD A,0 which needs two bytes.

 ;' It also handily clears the carry flag, which we'll be using.

 ;' We'll use the A register to build up our output.


 MirrorLoop: ;' This is a label we can jump to later. Sort of like a line number - it marks a point for the

  ;' assembler to say "remember this place with this name"

  RR C ;' This is our divide by two instruction - rotate right C. Instead of losing the last bit, though,

  ;' it is put into the carry flag.

  RLA ;' This does the A=A*2 part - it rotates the A register left. It also rotates in the carry flag on the

  ;' right, instead of a zero.

  ;' This means we're sliding bits off C one by one to the right into the carry flag, and onto A going left

  ;' from the carry flag. We're basicall using the flag as a temporary 1 bit store.

  DJNZ mirrorLoop ;' Decrease B and if it isn't zero, jump to mirrorloop.

  ;' So the A register now holds our mirrored result, and we can return it.

  RET
```

```
 END ASM
END FUNCTION
```

If you plug the above code into the first program, you can replace the ZX BASIC function with one that's hand assembled. The program should work identically, but be somewhat faster and smaller. We don't gain much speed, because most of the time isn't spent in the mirror function.

There are more examples of assembly code functions in the ZX BASIC wiki library at

http://www.boriel.com/wiki/en/index.php/ZX_BASIC:Library

The above code /isn't/ the most efficient in the world. For example, it would be possible, since we don't use the HL register for anything else, to not push the return address onto the stack, but to hold it in HL, and instead of a ret instruction at the end, which pulls an address off the stack and goes there, we could do jp hl instead.

Note that you can ONLY get away with ending a function with ret (or sneaky jp hl tricks) if it's a fastcall function. If it's not fastcall, there's housekeeping to do, so you should end the function by letting it reach the END FUNCTION statement. Most of the library functions aren't fastcall, so you'll see they often end by jumping to a label at the end of the machine code.

Britlion

http://www.boriel.com/forum/how-to-tutorials/tutorial-how-to-put-inline-assembly-functions-into-zx-basic-t729.html

Deutsche Zusammenfassung

In oben abgedruckten Artikel beschreibt Britlion wie man Assemblerfunktionen in Boriels ZX BASIC einbindet.

Hypothek 2

Hallo Spectrumfreunde, hallo Hans,

im Clubheft 226 (3. Quartal 2011) stellte Hans Schmidt ein altes BASIC-Listing zur Berechnung der Tilgungsraten für eine Hypothek vor und rief auf, Fehler zu suchen. Hier das Ergebnis: Die (schwierigen) Formeln zur Berechnung der Rate und der Restschuld sind völlig in Ordnung, fehlerhaft sind die ganz einfachen Formeln am Ende des Programms:

Zeile 320: Ersetze T=M*12*N

Zeile 350: Ersetze C=T/D.

Dann läuft alles prima. Solche Programme waren damals wirklich hilfreich, Mathcad, Maple oder Mathematica waren ja noch nicht erfunden!

Spectrale Grüße,

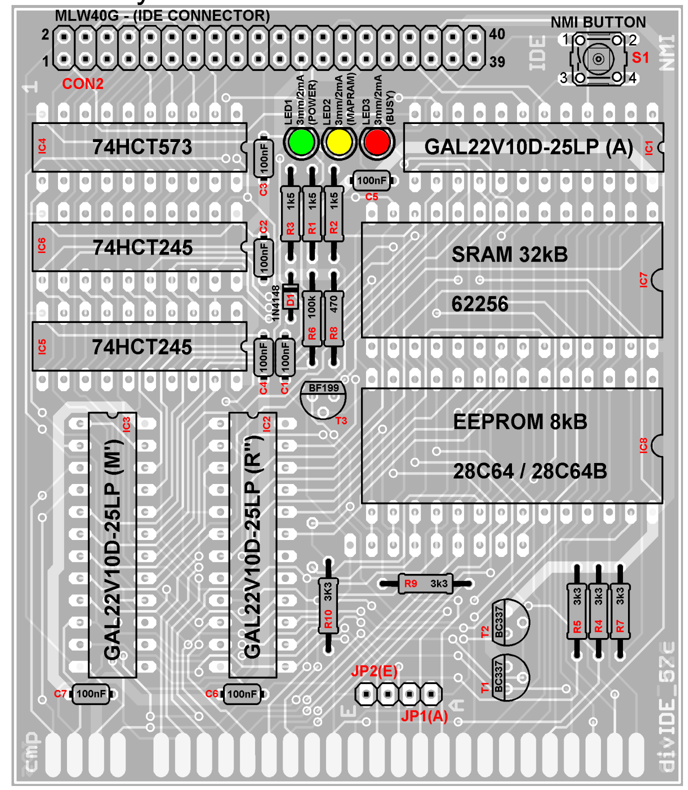
Heinz Junek.

DivIDE zum Selberlöten

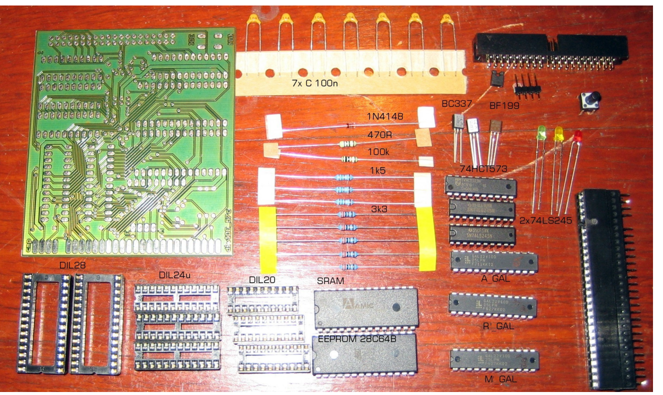
Spaßeshalber habe ich mir vor kurzem bei Noby ein DivIDE Set zum Selberlöten gekauft. Der sensationelle Preis von € 31,- für den Bausatz und € 15,- für SD-Adapter (stehend) haben mich verleitet das Set auszuprobieren, zudem da ich es mir in ein Gehäuse einbauen will, ist ein Bausatz leichter für meine Zwecke modifizierbar. Zugegeben, es gehört viel Arbeit dazu, aber es lohnt sich, deswegen habe ich gleich 2 Stück bestellt.



Ich war überrascht, dass so viele Vorarbeiten bereits erledigt wurden, so musste ich nicht den Spectrum-Connector selber sägen oder Key-Pin reinschmelzen, wie es in der Anleitung beschrieben ist.
Um meine DivIDEs unterscheiden zu können, habe ich bei diesem Modell einen weißen IDE-Stecker ausgestattet, das andere bekommt einen roten oder blauen. Im Original ist ein schwarzer dabei. Für mich ist es kein Problem weil ich solche Ersatzteile aus PC-Mainboards entlöte bevor sie am Mistplatz landen. Mit einem Heißluft-Lötkolben geht das relativ leicht, und PC Mainboards gehen bei mir oft ein.
Das Board das Noby verkauft, ist 2,0 mm dick und dadurch stabiler als z.B. das von Lotharek mit 1,4 mm, zudem sockelt Lotharek das EEPROM und SRAM nicht. In der Funktion ergibt sich dadurch zwar kein Unterschied, wenn man aber das EEPROM zu Tode beschreibt, ergibt sich dadurch enorme Mehrarbeit. Dabei muss ich erwähnen, dass EEPROMs nicht beliebig oft beschreibbar sind.
Das Layout des DivIDE ist einfach:



Alle Chips werden wie bereits erwähnt, gesockelt. Die Einkerbung zeigt dabei entweder nach Rechts oder nach oben.



Mit der Bauanleitung und allen notwendigen Informationen zum Aufbau des Interfaces auf http://divide.cz/index.php?y=kit.en ist es mit ein wenig Vorkenntnissen in Elektronik sehr einfach das DivIDE aufzubauen. Man sollte aber eine Lupe und ein Multimeter zur Hand haben um die Bezeichnungen auf Transistoren zu lesen sowie Widerstandswerte zu erkennen (da ich grün-blind bin, nützen mir Farbcodierungen herzlich wenig, und seit einem Eingriff bin ich nicht mehr in der Lage so kleine Schriftzeichen zu lesen).
Das EEPROM war mit FATWare 0.12a programmiert, was ich aber nicht bemerkte und mit einem EPROMER am PC nochmals einspielte nachdem es nicht funktionieren wollte. Nun, dass es nicht funktionieren wollte, lag an meinem Test-Speccy 48K+ der zum Testen der Hardware so umgebaut ist, dass die Spannungswandler-Transistoren gesockelt sind und sehr leicht ausgetauscht werden können. Auf der Forever treffe ich mich mit Noby und werde ihm mal das Problem demonstrieren. Auf dem Spectrum 128 funktionierte alles von Anfang an einwandfrei. Vielleicht liegt es am Z80, da einige Klone dieser CPU Probleme mit DivIDE haben. Ich werde dann noch einen Spectrum mit gesockelter CPU testen.
Zu den Modifikationen: Ich habe Pfostenstecker am NMI Button montiert, sowie statt den LEDs, damit kann man dann leichter das Interface bedienen wenn es im Gehäuse eingebaut wurde.

Nun bin ich dabei ein passendes Gehäuse zu suchen.
Für Bastlerfreaks ist es zu empfehlen, und man lernt dabei so einiges und hat auch noch Spaß wenn das DivIDE danach funktioniert.
Ich empfehle einen 25 Watt Lötkolben, oder einen der elektronisch steuerbar ist, zu benutzen.

LCD

Adventurelösung: The Curse of Nimue Teil 2

Hallo zusammen,

heute kommen wir also, wie angekündigt, zum zweiten Teil des Adventures **Curse of Nimue** von Karen Tyers. Wie ihr euch vielleicht noch erinnert, war unsere Aufgabe die Befreiung von Merlin aus Nimues Gefangenschaft und die Rückgabe aller Zaubersachen an Merlin. Hier im zweiten Teil kommt nun noch eine kleine Hürde hinzu. Zusätzlich zu der bekannten Aufgabe müssen wir noch Merlins Pokal (goblet) finden und an ihn zurückgeben um Nimues Fluch endgültig zu brechen. Außerdem ist das die Voraussetzung für unsere Rückteleportierung und die endgültige Möglichkeit, das Programm erfolgreich zu beenden. Jetzt haben wir uns aber wieder lange genug mit der Vorrede aufgehalten und es ist an der Zeit, wieder einen Blick auf den Plan und die Locations bzw. Gegenstände zu werfen, bevor wir uns ins Getümmel der Lösung stürzen:

01) At the bottom of a steep hill
02) Walking along a grassy slope / Nimue
03) Scrambling up a rough path
04) Still scrambling through rocks / serveral large rocks
05) Outside a temple / large heavy rock
06) Balancing unsteadily on top of the large rock / sign
07) Inside the temple / slabs, torches
08) Standing by the altar / woman, candles, wax, axe
09) In the middle of a small settlement
10) At the smithy / workbench, tools è hammer, bellows, fire, trough, pair of tongs
11) At the entrance to the catacombs
12) In the catacombs A
13) In the catacombs B
14) In the catacombs C
15) In the catacombs D / skeleton, bracelet
16) In the catacombs E
17) At a dead end / brick, hole, helmet, box, diamond
18) In a small room / inscription
19) In an archway wearing the helmet
20) Standing with the helmet on my head north of the bright light è walking along a north/south gravel path / trees, branch
21) At the top of a tree sitting among some very long branches / branch
22) On the south shore of a lake / small rowing boat
23) Sitting in a boat
24) Sitting in a boat at the middle of the lake / hand, sword
25) Sitting in a boat by the north shore of the lake
26) On the north shore of the lake
27) Among tall trees
28) Walking through a clearing
29) Entered a dimly lit cave
30) At the east end of the cave / rough wooden table, some stools, hook on the wall
31) Entered a secret room behind the cave / crystal goblet, pedestal,

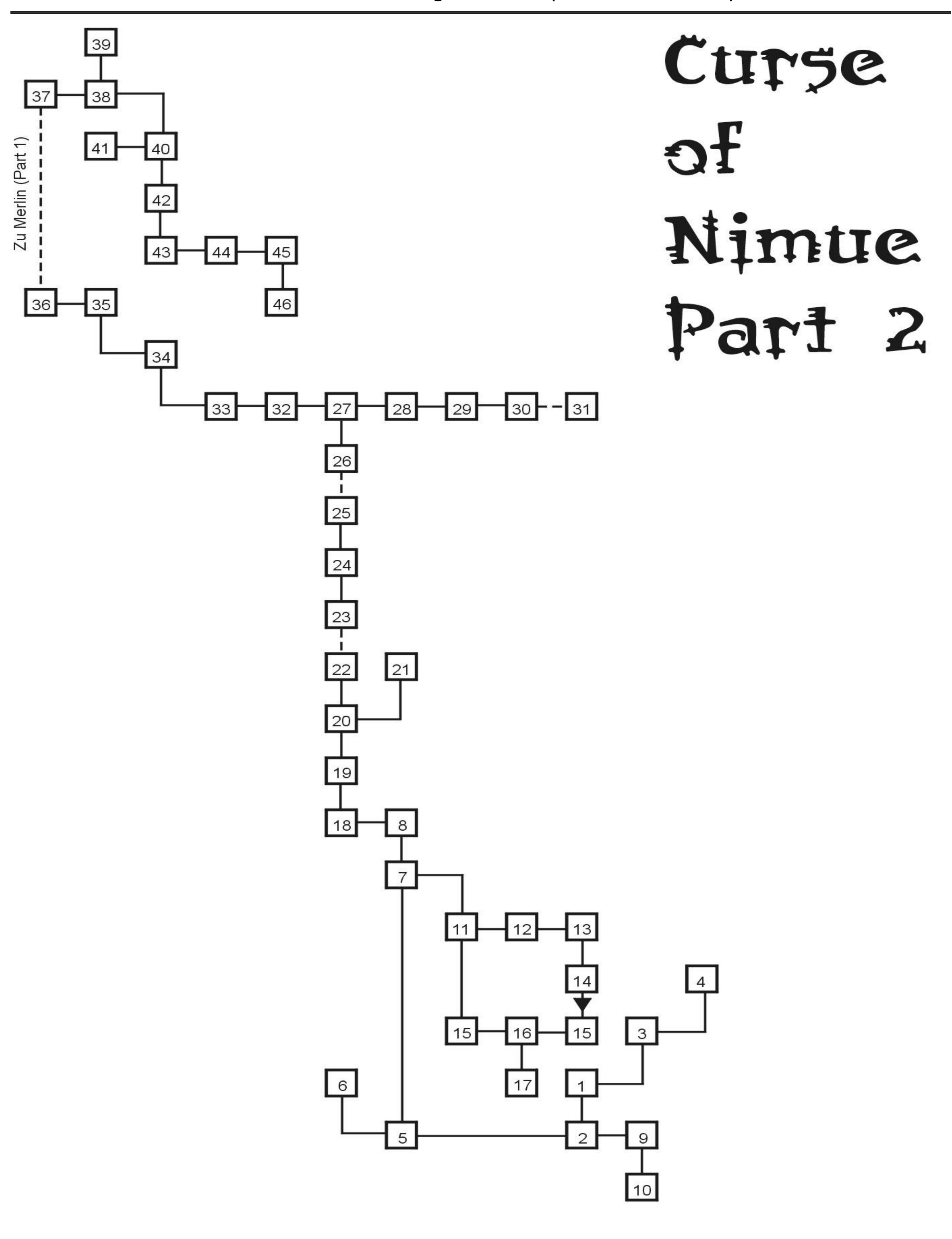
engraving
32) Walking along a path leading west / tall knight, chestnut
33) At the bottom of a very steep hill
34) Halfway up the hill
35) Standing on the summit beside the stone circle
36) Inside the stone circle
37) Back in my own bedroom / wardrobe, tracksuit, pair of trainers, pockets, 2 pound coin, bed
38) At the landing / picture
39) In my bathroom / mirror, bath, toilet, basin, medicine cabinet
40) In the hall at the bottom of the stairs / shelf, load of junk, screwdriver
41) In the lounge / sofa, armchairs, coffee-table, magazine, piece of paper, 2 pound coin
42) In my front garden
43) On the street
44) Further along the east-west street / gutter, storm drain, pound coin
45) In the village main Street
46) Inside the pharmacy / bottle

So, das waren sie also, alle relevanten Locations dieses Adventures und nun kommen wir auch gleich zur Lösung bzw. den Lösungshinweisen. Beginnend an Location 1 machen wir so weiter.....

Wir starten.... at bottom of steep hill. I (bag, bar und lance sowie scabbard), LOOK IN BAG (knife, trowel, horseshoe, gauntlet und jar), S (wir stoßen auf Nimue), E (geht nicht), W (geht auch nicht), N, U, U, X ROCKS, PUSH ROCK (den Hang runter), D, D, S (Nimue fragt uns nach dem Fels), PUSH ROCK W (direkt über ihren Fuß; wir befinden uns nunmehr direkt außerhalb eines Tempels), U (rauf auf den Felsen), READ SIGN (wir können jetzt den Tempel betreten), D, N, N (beim Altar), SAY HELLO, X CANDLES (wir bemerken, dass sie entzündet sind - da ist auch noch etwas Wachs), TAKE WAX, S, X SLABS, X BLUE SLAB (locker), LEVER BLUE SLAB (das Ende der Stange ist zu dick), S, E, E (mitten in einer kleiner Siedlung), S (die Dorfschmiede), X WORKBENCH, SEARCH TOOLS (wir entdecken einen Hammer), LOOK UNDER WORKBENCH (wir haben jetzt einen Blasebalg), X FIRE, PUMP BELLOWS (da ist ein kleines Loch drin - die Luft entweicht), PLUG HOLE WITH WAX, PUMP BELLOWS (jetzt glüht das Feuer richtig auf), TAKE GAUNTLET FROM BAG, WEAR GAUNTLET, HEAT BAR (wir müssen den Schutzhandschuh tragen), FLATTEN BAR (wir brauchen dazu den Hammer), LOOK BEHIND TROUGH (wir haben jetzt eine Zange), TAKE HORSESHOE FROM BAG, PUMP BELLOWS, HEAT HORSESHOE (braucht die Zange), FLATTEN HORSESHOE (mit dem Hammer), DROP HAMMER, DROP TONGS, DROP BELLOWS, N, W, W, N (im Tempel), LEVER BLUE SLAB WITH BAR (muss ein angeflachtes Ende haben - wir entdecken eine Treppe, die nach unten führt), DROP BAR, X TEMPLE (da sind Fackeln), TAKE TORCH, X TORCH (nicht angezündet). N, LIGHT TORCH (mit Hilfe der Kerze), S, D (der Eingang zu den Katakomben), E, E, S, S (hier ist ein Skelett), SEARCH SKELETON (jetzt haben wir ein Armband), X BRACELET, W, S (Sackgasse), X BRICK (einer ragt heraus), PUSH BRICK (da ist jetzt ein Loch in der Mauer), FEEL, IN HOLE (hier finden wir einen Helm), X HELMET, TAKE TROWEL, FROM BAG, DIG (wir

finden eine Schachtel), DROP TROWEL, TAKE BOX, OPEN BOX (wir finden einen Diamanten). X DIAMOND, N, W, N, U (zurück im Tempel), N, GIVE DIAMOND (wir erhalten eine Axt und eine Tür im Westen öffnet sich), W (die Fackel verschwindet - aber natürlich brauchen wir sie auch nicht mehr), READ INSCRIPTION, WEAR HELMET, N, N, REMOVE HELMET, DROP HELMET, X TREES, CLIMB TREE (wir sollten uns den Ausblick gut einprägen), D, N (das Ufer eines Sees), ENTER BOAT, X BOAT (keine Ruder hier), LEAVE BOAT, S, CLIMB TREE, CHOP BRANCH (nur mit unserer Axt), D, CHOP BRANCH (in zwei Teile), DROP AXE, TAKE BRANCHES, TAKE KNIFE FROM BAG, CARVE BRANCHES (wir machen mit dem Messer daraus Ruder), DROP KNIFE, N, ENTER BOAT, PUT OARS INTO ROWLOCKS, ROW N, ROW N (eine Hand greift aus dem Wasser und stoppt unser Boot), X HAND, X MARK, THROW BRACELET AT HAND, TAKE SWORD, X SWORD, X STONES (Smaragde), ROW N, LEAVE BOAT, N, E, E, E (wir suchen das Ende der Höhle), SEARCH CAVE (da ist ein Haken an der Wand), X HOOK, TURN HOOK (nichts scheint zu passieren), SAY UTHER (wir gelangen in einen Geheimraum - wenn wir den Haken gedreht haben, sollte hier ein Pokal zu sehen sein, falls nicht, ist er noch von einem Schutzfeld umgeben und unsichtbar), X PEDESTAL, X ENGRAVING (Waage), TAKE JAR FROM BAG, X JAR (Anleitung merken!!), SWAP GOBLET WITH JAR, X GOBLET, SAY UTHER, W, W, W, W (wir treffen auf einen Ritter), X KNIGHT, SAY HELLO, GIVE LANCE, REMOVE SCABBARD, GIVE SCABBARD, GIVE SWORD, REMOVE GAUNTLET, GIVE GAUNTLET, GIVE HORSESHOE (nur wenn er geplättet ist), W, U, U, W (das ist okay, wenn wir den Pokal dabei haben), SAY VAPOURISE (wie es auf der Seite des Buches in Teil Eins stand - wir sind jetzt zurück in der Höhle, in der Merlin als Gefangener gehalten wird), GIVE GOBLET (wir gelangen, wie versprochen, wieder nach Hause).

Wir sind jetzt wieder in unserem Schlafzimmer.... OPEN WARDROBE, REMOVE ALL (alles verschwindet), TAKE ALL (wir haben jetzt einen Trainingsanzug und Turnschuhe), WEAR ALL, X TRACKSUIT (hat Taschen), SEARCH POCKETS (da ist eine 1 Pfund Münze), X BED (nicht gemacht), MAKE BED (irgendwas fällt auf den Fußboden), LOOK UNDER BED (wir haben jetzt zwei Pfundmünzen), E, X PICTURE, N, X MIRROR. S, D (die Halle), X SHELF, SEARCH JUNK (jetzt haben wir einen Schraubendreher), W (die Diele), X TABLE (da ist ein Magazin), TAKE MAGAZINE, X MAGAZINE, READ MAGAZINE (ein Stück Papier fällt heraus), DROP MAGAZINE, TAKE PAPER, READ PAPER (Rezept), X SOFA (da ist ein Loch im Polster), FEEL IN HOLE (jetzt haben wir drei Pfundmünzen), SEARCH ARMCHAIR (da ist die vierte Pfundmünze), E, S, S, E (auf der Straße), X GUTTER, X DRAIN, UNSCREW DRAIN (mit Hilfe des Schraubendrehers), DROP SCREWDRIVER, SEARCH DRAIN (fünf Pfundmünzen), E, S (Apotheke), SAY HELLO, GIVE PRESCRIPTION, GIVE COINS (wir erhalten eine Flasche), X BOTTLE, READ LABEL, DRINK CONTENTS.


Curse of Nimue Part 2
Zu Merlin (Part 1)
39
37 38
41 40
42
43 44 45
36 35 46
34
33 32 27 28 29 30 31
26
25
24
23
22 21
20
19
18 8
7
11 12 13
14 4
15 16 15 3
6 17 1
5 2 9
10

100/100

Das war alles - The Curse of Nimue ist gelöst. Wollen wir mal sehen, was wir als Nächstes ausgraben. Lasst euch also überraschen. Bis dahin viel Spaß mit dem Spectrum.

English Summary:

Dear users,

as we have promissed in the last issue of the club magazine, we come today to the second and last part of “The Curse of Nimue“. As you all know from the first part of the solution, our task is to find all of Merlin´s magical instruments and free him from the clutches of the evil witch Nimue. But at this point a new challenge has appeared. We must find an return the goblet of Merlin to break the spell of Nimue at last. This is necessary for us to be returned to our normal life at the end of the asdventure. But we won´t stop you from playing the second part. In case of trouble use our solution an map if necessary.

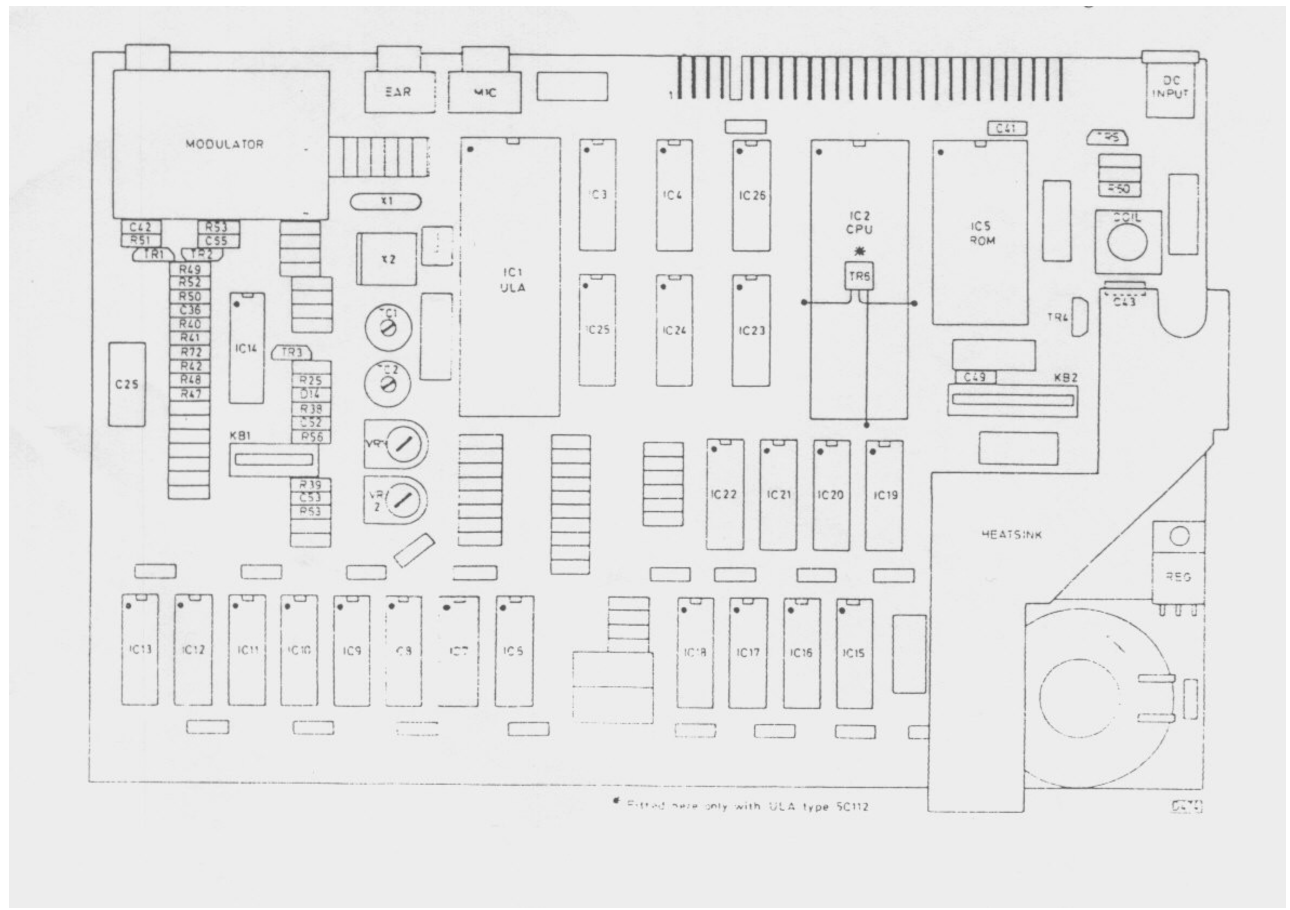
Servicing Sinclair Computers Pt. 5

http://www.worldofspectrum.org/hardware/rep6.html

Previous treatment of the Spectrum in this series has related specifically to the issue 3 and 3B versions. There has however been continuous development of the machine since its introduction in 1982. The range now extends from the initial issue 1 version to issue 6A, covering a total of eight models with six PCB changes. It's time we looked at some of these variants, starting with the earlier models.

The issue 1 and 2 versions have a lot in common. They both have the early zig-zag shaped heatsink, and both have small trimmer capacitors and preset potentiometers for setting up the colour generator circuits. The issue 2 board layout is shown in fig. 8. The issue 1 differs only in the design of the original 32K extension memory, which isn't assembled on the main board as with all later versions. Instead, its built on a plug-in board that carries the memory ic's and the decoders and multiplexers. This daughter board plugs into two DIL sockets at the rear of the main board and extends right across from the modulator to the coil. To accommodate this, the CPU and the multiplexer chips IC3/4 are moved towards the front of the main board along with the ULA and ROM chips, leaving a clear space on which the extension memory board fits. If this space isn' t filled the machine isn' t worth very much, since the extension is no longer available and without it the majority of the commercial programs cannot be used.

Another distinctive feature you will find on some Issue 1 boards is the "spider". Due to a timing error in the 5C102 ULA chip it was necessary to fit an extra 74LS(X) i.c. Because this was added retrospectively there wasn' t room for it on the board, so the i.c. had to be mounted on its own small board suspended above the main board by the connecting wires. When the later type 5C112 ULA was introduced the spider was no longer required. The initial issue 2 board used the same 5C112 ULA but a



Layout of the issue 2 board

further modification was fitted: this was the addition of TR6 which replaced the previous diode/resistor network - see Fig. 9. As you can see from Fig. 8. TR6 is mounted across the top of the CPU i.c. Later versions of the issue 2 board have the current 6C001 ULA chip: this necessitated some resistance changes which are detailed in (4) of the issue 2 modification instructions.

The following modifications should be added whenever a Spectrum is dismantled for servicing. First. issue 1 versions.

Issue 1 Modifications

(1) When National 4116 RAM i.c.s are fitted, remove R57 (330ohm) - connected to pin 28 of IC2 - and fit a 1kohm resistor between the CAS line and the 12v rail and another 1kohm resistor between the RAS line and the 12v rail. These resistors are best fitted on the underside of one of the memory chips IC6-13. C54 (at pin 3 of IC2) can also be removed - but it must be left in circuit when the 4116 RAMs are of NEC manufacture.

(2) When a type 5C102 ULA is fitted. add a 100pF capacitor between the RAS line and chassis.

(3) C46 (1uF electrolytic) should be replaced with a high-temperature capacitor it' s mounted beneath the heatsink.

(4) Axial capacitors should be fitted in place of all the disc ceramic capacitors. The following capacitors must he replaced: C41 (ROM pin 14 to pin 28) and C49 (between the collector and

emitter of TR4) - these capacitors are both 47nF.
(5) If there' s insufficient colour difference between white and yellow, fit a 47kohm resistor between pin 13 of IC14 (LM1889) and chassis.
(6) To improve the reliability of the voltage generator the circuit should be modified to correspond with Fig. 13. A minimum would be to change the value of R60 and fit a 4.7uF electrolytic (C74) between the emitter and base of TR5. See the notes on this section of the circuit in Part 6 next month.
(7) Finally, if you want to use the Spectrum to operate a Z80 PIO, or if you find that some machine code software doesn' t run satisfactorily, check that the following modifications have been made.
(a) Change TR3s base circuit as shown in Fig 10. i.e. replace D14 with C67, change R24 to Ikohm and add the pull-up resistor R73.
(b) Change R27 from 680ohm to 47oohm or shunt it with a I.5kohm resistor (ULA pin 33 to CPU pin 20).

Issue 2 Modifications

Now to issue 2 boards. Like the modifications given for the issue 1 version these should be made whenever possible. Fig. 8 shows the positions of the components.
(1) Replace all disc ceramic capacitors with axial ones. Especially change C41 an d C49 (47nF) - as with issue 1 boards - and change C43 (100nF) in the voltage generator circuit. This, together with modification (3) below will update the circuit almost to issue 3 standard.
(2) To improve the colour, change R48S to 2.2kohm. R49 to 8.2kohm. R50 to 4.7kohm. R72 to 10kohm and C65 to 22uF. These components are all associated with the luminance/chrominance drives to TR1 and TR2 (see Fig. 11).
(3) Carry out the same modifications as those listed under (6) and (7) for issue 1 boards.
(4) The only currently available ULA is type 6C001. When this is used to replace an earlier type the following modifications should be made: change R47 to Ikohm. R49 to 10kohm and R56 and R63 to 470ohm.

(5) There' s no need to change the speaker circuit from that shown in Fig. 5 to that shown in Fig. 11. The modification is very simple however if increased sound Output is require.

(6)

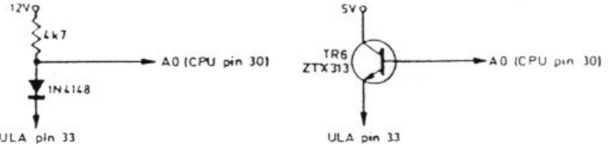
Servicing Aspects

From the servicing point of view the advice given to issue 3 versions applies in general to issue 1 and 2 versions. There' s one exception. There are four presets- (TC1. TC2, VRI and VR2) that may need setting up if any changes have been made. Their positions are show in Fig. 8 and their functions are as follows.
TC1 sets the frequency of the 14MHz crystal that controls all the computer timing, including the 50Hz field sync signal. You might think that this would provide an easy means of setting up this control, but in many cases the range of adjustment is too small to enable the 50Hz to be locked. The control is used only to alter the frequency slightly, to eliminate any objectionable colour patterning on the screen.
TC2 sets the frequency of the colour subcarrier oscillator and unlike TC1 Sinclair advise precise adjustment using a frequency counter. I' ve personally had no problems with the setting of this

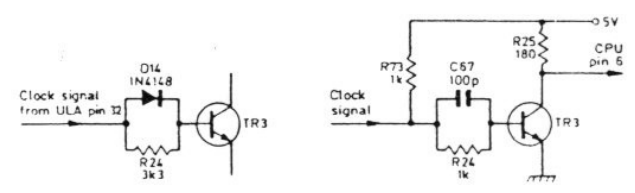
control but if a check is required it should be possible to compare the results with the frequency obtained from a TV set locked to a transmitter.
VRI and VR2 are the only controls that may present difficulties. They affect the phasing of the colour-difference signals and are interactive in their effect on the display. Take particular care when dealing with issue 1 models because although the controls are in the same positions and are marked as shown in Fig. 8, the connections between them and IC14 are reversed. A colour display is necessary for setting and checking these controls. The following short program will display colour bars, enabling the effect of any changes to be seen across the colour spectrum. It' s advisable to sav e this program to make it easier to load when the top case and keyboard are lifted to reach the presets.

```
10 FOR N = 0 T0 7
20 FOR M = 0 TO 3
30 PAPER N : PRINT " "
40 NEXT M
50 NEXT N
60 GOTO 10
```

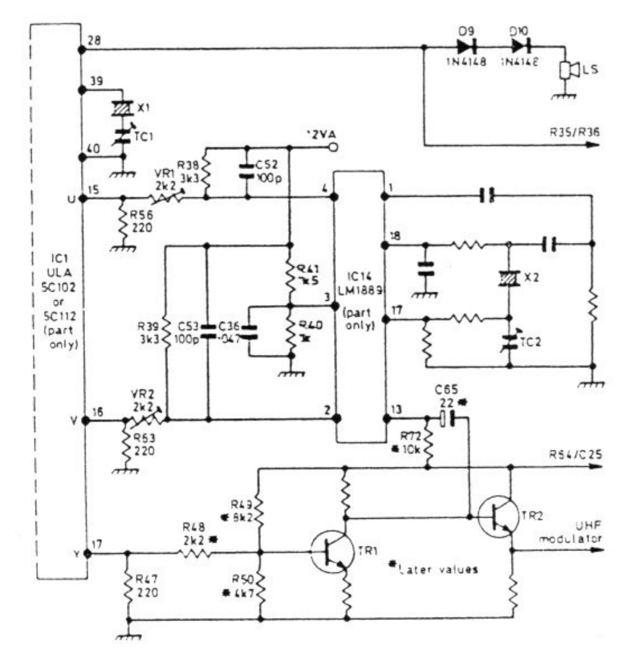
This will display the Spectrum colours corresponding to keys 0 to 7. ie. black, blue, red, magenta, green, cyan, yellow and white.
If there is no colour on the screen when this program is entered and run, check the TV sets tuning and colour controls. If there' s still no colour the controls will have to be set up. The procedure suggested by Sinclair is as follows.
(1) Switch on and initialise the computer. Do not enter a program.
(2) Using TC2. set the colour subcarrier frequency to 4.433619MHz +/- 50Hz
(3) Using VR1. set the voltage at pin 4 of IC14 to 50rnV + 0mV/-5mV relative to pin 3.
(4) Using VR2. set the voltage at pin 2 of IC14 to -50mV +5mV/-50mV relative to pin 3.
These settings are designed so that pins 2 and 4 will be at zero with respect to pin 3 when the computer is at its operating temperature. In the factory however they set pin 4 to 130mV +/- 20mV and pin 2 to -75mV +/- 20mV, so you can take your choice which values to use.



Modifications associated with pin 33 of IC1: early version left, later version right.



Modified clock circuit. Its esential to modify the circuit to that shown on the rightwhenever the earlier version is found.



Circuit variations around IC1/IC14 in earlier versions - compare with figs 1 and 2.

Personally I prefer the following method of setting these controls. It may seem very complex at first sight but it's actually quite simple. A word of explanation. Those of you who are long in the tooth - and short of hair - may remember ion traps. These could be set in a few seconds but it took you twice as long even to read the Mullard instructions. This procedure is similar. As the settings aren't critical - about the same as the average hold control - getting the colour correct is easier than reading the instructions. So here they are:

(1) Load the program above and run it.

(2) Assume that the subcarrier frequency is o.k. and set VR1/2 to mid-travel.

(3) Slowly sweep VR1 until colour is displayed. If no colour shows during the full travel of VR1 , move VR2 slightly and try again.

(4) Keep moving VR2 in steps of about 20-30 degrees, sweeping VR1 slowly back and forth until colour has been obtained, or the whole range of both potentiometers has been covered.

(5) If no colour can be obtained at any settings, mark the position of the vanes of TC2 and move it approximately 30 degrees. Repeat (3) and (4) above. At worst it should take only about four- five repeats to get some indication of colour.

(6) When some colour is displayed, move the presets one at a time until the full eight colour bars are present, in their correct colours. Finally find the optimum position for each adjustment, going over TC2, VR1 and VR2 at least twice.

The colour controls should now be set up correctly and the bars displayed in their correct colours. Check by switching channels on the TV set and making sure that the colours lock without any delay. When all is well set TC1 as described below.

If you still have a problem there could be a fault in IC14, the associated circuitry or the signals from the ULA. The colour-difference signals can be checked either at pins 17 and 18 of the edge connector (underside) or at VR1 and VR2. Examination of the signals with and without the colour display running will show if the ULA is o.k. Check the oscillator and its frequency at pin 17 of IC14 - use a high-impedance probe when checking the frequency.

Finally, when a satisfactory colour display has been obtained put in a program giving a screen full of characters in red ink with a background of green paper and adjust IC1 for minimum patterning. Some early machines have a hole in the bottom of the case to enable this adjustment to be carried out with the computer fully assembled.

Next month well deal with the 4A, 4B, 5 and 6A versions.

Jim Grimwood

Deutsche Zusammenfassung:

Im vorletzten Teil der Reihe beschreibt Jim die Videoausgabe des Spectrum und wie sie justiert wird wenn der Bildschirm Farben schlecht darstellt.

Auch Veränderungen zwischen den verschiedenen Ausgaben der Spectrum-Platinen werden hier näher beleuchtet

Vorschau auf die Ausgabe 229

Nächstesmal werde ich die Anleitung der **256 Kb Speichererweiterung** für Spectrum 16/48 K veröffentlichen.

Außerdem stelle ich das **Trinity** für SAM Coupé vor.